The application of the cross-entropy method for multi-objective optimisation to combinatorial problems

Charlotte Hauman

Thesis presented in partial fulfilment of the requirements for the degree of Master of Science in the Faculty of Engineering at Stellenbosch University

Study leader: J. Bekker

Date: December 2012

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 1 September 2012

Abstract

Society is continually in search of ways to optimise various objectives. When faced with multiple and conflicting objectives, humans are in need of solution techniques to enable optimisation. This research is based on a recent venture in the field of multi-objective optimisation, the use of the cross-entropy method to solve multi-objective problems. The document provides a brief overview of the two fields, multi-objective optimisation and the cross-entropy method, touching on literature, basic concepts and applications or techniques. The application of the method to two problems is then investigated. The first application is to the multi-objective vehicle routing problem with soft time windows, a widely studied problem with many real-world applications. The problem is modelled mathematically with a transition probability matrix that is updated according to cross-entropy principles before converging to an approximation solution set. The highly constrained problem is successfully modelled and the optimisation algorithm is applied to a set of benchmark problems. It was found that the cross-entropy method for multi-objective optimisation is a valid technique in providing feasible and non-dominated solutions.

The second application is to a real world case study in blood management done at the Western Province Blood Transfusion Service. The conceptual model is derived from interviews with relevant stakeholders before discrete event simulation is used to model the system. The cross-entropy method is used to optimise the inventory policy of the system by simultaneously maximising the combined service level of the system and minimising the total distance travelled. By integrating the optimisation and simulation model, the study shows that the inventory

policy of the service can improve significantly, and the use of the cross-entropy algorithm adequately progresses to a front of solutions.

The research proves the remarkable width and simplicity of possible applications of the cross-entropy algorithm for multi-objective optimisation, whilst contributing to literature on the vehicle routing problem and blood management. Results on benchmark problems for the vehicle routing problem with soft time windows are provided and an improved inventory policy is suggested to the Western Province Blood Transfusion Service.

Opsomming

Die mensdom is voortdurend op soek na maniere om verskeie doelwitte te optimeer. Wanneer die mens konfrontreer word met meervoudige en botsende doelwitte, is oplossingsmetodes nodig om optimering te bewerkstellig. Hierdie navorsing is baseer op 'n nuwe wending in die veld van multi-doelwit optimering, naamlik die gebruik van die kruis-entropie metode om multi-doelwit probleme op te los. Die dokument verskaf 'n breë oorsig oor die twee velde – multi-doelwit optimering en die kruis-entropie-metode – deur kortliks te kyk na die beskikbare literatuur, basiese beginsels, toepassingsareas en metodes. Die toepassing van die metode op twee onafhanklike probleme word dan ondersoek. Die eerste toepassing is dié van die multi-doelwit voertuigroeteringsprobleem met plooibare tydvensters. Die probleem word eers wiskundig modelleer met 'n oorgangswaarskynlikheidsmatriks. Die matriks word dan deur kruis-entropie beginsels opdateer voor dit konvergeer na 'n benaderingsfront van oplossings. Die oplossingsruimte is onderwerp aan heelwat beperkings, maar die probleem is suksesvol modelleer en die optimeringsalgoritme is gevolglik toegepas op 'n stel verwysingsprobleme. Die navorsing het gevind dat die kruis-entropie metode vir multi-doelwit optimering 'n geldige metode is om 'n uitvoerbare front van oplossings te beraam.

Die tweede toepassing is op 'n gevallestudie van die bestuur van bloed binne die konteks van die Westelike Provinsie Bloedoortappingsdiens. Na aanleiding van onderhoude met die relevante belanghebbers is 'n konsepmodel geskep voor 'n simulasiemodel van die stelsel gebou is. Die kruis-entropie metode is gebruik om die voorraadbeleid van die stelsel te optimeer deur 'n gesamentlike diensvlak van die stelsel te maksimeer en terselfdetyd die totale reis-afstand te minimeer. Deur

die optimerings- en simulasiemodel te integreer, wys die studie dat die voorraadbeleid van die diens aansienlik kan verbeter, en dat die kruis-entropie algoritme in staat is om na 'n front van oplossings te beweeg. Die navorsing bewys die merkwaardige wydte en eenvoud van moontlike toepassings van die kruis-entropie algoritme vir multi-doelwit optimering, terwyl dit 'n bydrae lewer tot die afsonderlike velde van voertuigroetering en die bestuur van bloed. Uitslae vir die verwysingsprobleme van die voertuigroeteringsprobleem met plooibare tydvensters word verskaf en 'n verbeterde voorraadbeleid word aan die Westelike Provinsie Bloedoortappingsdiens voorgestel.

Acknowledgements

- Mr. James Bekker for providing the inspiration for the research, for providing Pareto optimal guidance during the past two years in all aspects and for being the best definition of a teacher.
- The International Offices of the University of Stellenbosch and the Vrije University of Amsterdam for enabling me, both logistically and financially, to complete a semester of study on exchange and become a student of life and the world.
- The Department of Industrial Engineering for providing the structures and necessary financial support to complete my degree(s).
- Ms. B. Alexander, Mr. D. Anderton and the rest of the staff at the WPBTS for their support and willingness to assist.
- Ms. Karen Hauman for the proofreading of this document and other linguistic advice.

- To my parents for their unconditional love and support in everything I dare to attempt and dream.
- To my sisters and friends for the emotional support during the past two years and for bearing with me when necessary.
- My Creator and Saviour for the ability to study, for proving His faithfulness when the rest of my comfort zone is far away and for countless blessings in my life. All the glory to Him!

I am able to do all things through Him who strengthens me.
Phil. 4:13

Financial Assistance

The partial financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the NRF.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Roman Symbols

sl_i	Average service level at blood bank i
c_1	Transport cost per unit per kilometer
c_2	Vehicle cost per kilometer travelled
d_{ij}	Cost and\or distance to travel arc i to j
N_m	Maximum number of loops used in the multi-objective cross-entropy algorithm
P	Probability distribution matrix of discrete optimisation
Q_k	Capacity of a vehicle in the vehicle routing problem
s_{ik}	Time at which vehicle k starts service at customer i
SL	Total service level of simulated WPBTS system
t_d^k	Total delay time of customers on a route waiting for vehicles that arrive after the close of a time window
t_h	Preset threshold value against which rank value of multi-objective solution vector is compared
t_w^k	Total time vehicle k waits for the start of time windows on a route
t_{ij}	Time to travel arc i to j

TC	Total cost of travel of simulated WPBTS system
TD	Total travelled distance of WPBTS system
D_i	The demand at customer i
I_H	Hyperarea indicator
l	Rare-event probability in importance sampling
N	User-specified population size for population-based algorithms
n	Number of customers in the vehicle routing problem
Z1	Vehicle routing problem objective: Number of vehicles
Z2	Vehicle routing problem objective: Total travel distance
Z3	Vehicle routing problem objective: Makespan of tasks
Z4	Vehicle routing problem objective: Total vehicle waiting time
Z5	Vehicle routing problem objective: Total vehicle delay time

Greek Symbols

γ	Value of sample of the upper quantile of objective functions used in the cross-entropy method
ρ_q	Rank value of multi-objective solution vector q
τ	Maximum number of evaluations per loop used in the multi-objective cross-entropy algorithm
ϱ	Value that determines size of quantile used in the cross-entropy method.
α	Smoothing parameter for the cross-entropy method

Other Symbols

$\mathcal{C}$	Set of customers in the vehicle routing problem
$\mathcal{N}_i$	Number of products at blood bank i

$\mathcal{O}_j$	Size of order of product j
$\mathcal{V}$	Set comprising a fleet of vehicles in the vehicle routing problem
I_j	Current inventory level of product j
$\mathcal{D}(g, f)$	Kullblack-Leibler divergence of f from g
$\mathcal{P}^*$	Pareto optimal set
$\mathcal{P}_T^*$	Pareto front

Acronyms

$f(x)$	Performance or objective function on x
ASP	Associated stochastic problem used in the cross-entropy method
CEM	Cross-entropy Method
COP	Combinatorial Optimisation Problem
DC	WPBTS Central Distribution Centre
GA	Genetic Algorithm
GEO	George blood bank
GSH	Groote Schuur hospital blood bank
LDRC	Leukocyte depleted red cells product
MCV	Mediclinic Vergelegen blood bank
MOEA	Multi-objective evolutionary algorithm
MOO	Multi-objective Optimisation
MOP	Multi-objective Problem
NSGA	Non-dominated Sorting Genetic Algorithm
PAARL	Paarl blood bank
RCC	Red cell concentrate product
RCX	Red Cross Children's hospital blood bank
TBH	Tygerberg hospital blood bank

TSP	Travelling Salesperson Problem
MOGA	Multi-objective Genetic Algorithm
VEGA	Vector Evaluated Genetic Algorithm
VRPSTW	Vehicle Routing Problem with Soft Time Windows
VRPTW	Vehicle Routing Problem with Time Windows
VRP	Vehicle Routing Problem
WB	Whole blood product
WPBTS	Western Province Blood Transfusion Service
WTR	Worcester blood bank

CHAPTER 1

INTRODUCTION

1.1 Background of the study

The world that we live in has become increasingly competitive. Enterprises are preoccupied with delivering better products, faster service and higher profits. Optimisation, the science of better, is an integral part of operations. In this drive to become better we are constantly confronted with having to make decisions that will maximise or minimise certain objectives while staying within certain boundaries or constraints. Objectives cover a wide range of aspects and often need to be optimised simultaneously. The human mind, regardless of exceptional logic and analytical abilities, is easily overwhelmed by the availability of various options especially when a number of goals have to be attained. It is often difficult to distinguish between seemingly conflicting objectives and the effect that decisions will have thereon. To this end the field of multi-objective optimisation was developed to deal with the presence of two or more conflicting objectives. The main aim is to assist the decision maker with an idea of the influence variables have on the feasibility of solutions and objectives and to display the interaction of multiple objectives.

One of the best examples of multiple objectives is an enterprise wanting to minimise cost while maximising the quality of their product. The quality of the product or service is in most cases directly linked to the amount of resources, the quality of the resources, the human capital required and the time that is allocated to the product or service. All of these factors affect the running costs of the enterprise and the team is faced with having to find an acceptable midway between these two objectives.

The field of multi-objective optimisation has been firmly established with scholarly research covering a wide range of subjects such as problems, ranking methods, performance measures and solving techniques. In the latter case Bekker & Aldrich (2010) document evidence that the cross-entropy method can be successfully applied to multi-objective optimisation. It is this research that provided the origin of the study presented in this document.

In the comprehensive reference on evolutionary multi-objective optimisation, Coello Coello (2009) discusses research trends that include new algorithms, efficiency, relaxed forms of dominance, scalability and alternative meta-heuristics. He further states that researchers propose new algorithms but only some become widely used. Bekker & Aldrich (2010) adapted the cross-entropy method to propose a new algorithm and provided results on well-known multi-objective test problems and the simulation of an inventory problem. Further research to cement the use of the cross-entropy method for multi-objective optimisation was needed. It is assumed that initial research on a new algorithm is essential to ensure that its worth is proclaimed. This led to the problem statement as addressed by this research.

1.2 Problem statement

The use of the cross-entropy method for multi-objective optimisation has been suggested in recent literature and preliminary tests on benchmark problems showed favourable results. The challenge is to find more evidence of the method in the field of multi-objective problems by applying the algorithm to a variety of problems. This challenge is scoped to a few combinatorial problems, including the academic vehicle routing problem with soft time windows (VRPSTW) and a real world problem in blood supply chain management as observed at the Western Province Blood Transfusion Service (WPBTS).

1.3 Research purpose

Apply the cross-entropy method for multi-objective optimisation to combinatorial multi-objective problems.

With the ultimate research aim in mind, general objectives for the study are defined below and the attainment thereof is documented in the following chapters. The purpose of the research is to investigate the use of the cross-entropy method for multi-objective optimisation on combinatorial problems. The intention is not to compare the method to other heuristics, but rather to illustrate the simplicity and worth of the method. The main objectives of the study are listed below:

- Study the literature on the field of multi-objective optimisation, acquiring a grasp of the relevant methodology, concepts and foundations as needed for the study. (**Chapter 2**)
- Study the literature on the cross-entropy method, including its applications, definitions and mathematical foundations. (**Chapter 3**)
- Understand the application of the cross-entropy method to multi-objective optimisation. (Bekker & Aldrich, 2010)
- Do a literature study and build a model of the multi-objective vehicle routing problem to enable the application of the cross-entropy method and document and interpret results. (**Chapter 4**)
- Overview the literature and do a real world case study of blood supply chain management in the context of the Western Province Blood Transfusion Service. Build a simulation and optimisation model of the problem to enable the application of the cross-entropy method and document and interpret results. (**Chapter 5**)
- Master the use of software packages LaTeX, Matlab® and Simio® to enable the previous objectives.
- Draw a conclusion on the worth of the multi-objective cross-entropy method in solving combinatorial problems. (**Chapter 6**)

With these broad objectives in mind the study was initiated. The refined objectives of the application of the method is listed in the separate chapters on the problems, Chapters 4 and 5. The definition of objectives provides a view of the methodology followed to obtain them as explained in the next section.

1.4 Methodology

The study originated in the application of the cross-entropy method to multi-objective optimisation as presented by Bekker & Aldrich (2010). This research showed favourable results with regard to this application and further applications to more problems were needed to proclaim the worth of the method. The first application was done to a largely theoretical problem, namely the vehicle routing problem with soft time windows (VRPSTW) to display the use of the method in highly constrained routing as a combinatorial problem. The second application is to a real world problem of inventory management in blood transfusion supply chain management.

A combination of mathematical modelling and computer simulation methods are used. Muller (2008) defines mathematical modelling as "building of a model based on theory and observation and predicting its performance on mathematical equations." In the case of the first application, the mathematical model of the vehicle routing problem was used in a computer simulation environment to conduct experimental tests. The model is based on basic equations of the VRPSTW as found in literature, whilst the cross-entropy method is modelled with transition probability matrix as explained in Chapter 4. Validation was done on smaller instances of the problem and by sampling a solution and ensuring that constraints are met. In addition, the research methodology consists of experimental research; "the isolation of a variable, whilst controlling the factors that may influence it", Muller (2008). This was done to determine good values for parameters used in the cross-entropy method.

The methodology followed to produce Chapter 5 can be summarised to a case study and simulation modelling. The case study was conducted by interviews with the relevant stakeholders and acquiring the data needed from the real world system. From this a conceptual model and simulation model were built and validated. Mathematical modelling was used to build the optimisation model. Finally, experimental research was again conducted.

1.5 Structure of the document

The structure of the document is founded in the defined objectives and formed by the methodology explained in the previous sections. This chapter introduces the field of multi-objective optimisation and the development of the research aim and objectives.

In **Chapter 2** an overview of the literature and advances in the field of multi-objective optimisation (MOO) is presented to provide insight on the origin of the problem statement and the field of research. The methodology and basic concepts of the field as needed for the rest of the document are explained to provide a backdrop on which the study was done.

This overview of the MOO field is then narrowed in **Chapter 3** with a discussion of the cross-entropy method (CEM). The theoretical foundations are explained, the use of the method for combinatorial optimisation and finally the application of the method to multi-objective problems which is the focus of this study.

With the outline of the study documented in the first few chapters, the next chapters serve to describe the details of the study. After the CEM for MOO is introduced and basic concepts are explained, the reader will be presented with the application of this method to combinatorial problems to show the processes followed to aid in the fulfillment of the research purpose.

Chapter 4 documents the application of the CEM for MOO to the vehicle routing problem with soft time windows. The chapter consists of a set of literature pertaining to the problem, the modelling of the problem and concludes with a discussion of the results.

In **Chapter 5** a case study on the inventory management of a local blood transfusion service is presented. A brief literature study on supply chain management and inventory management in the context of blood transfusion serves as an introduction to the centre. The conceptual model, simulation model and optimisation via the CEM are documented before the chapter is concluded with a discussion of the results.

Finally **Chapter 6** provides a conclusion to the document. The final interpretations and suggestions for future research are listed. The reader is also referred to the appendices which contain the bulk of the experimental results and other additional material. Specific references will be made throughout the document where necessary.

CHAPTER 2

MULTI-OBJECTIVE OPTIMISATION

Society is constantly in search of ways to do things faster, cheaper or simply better. Decisions are made on a daily basis in various sectors and life in general in the hope of obtaining good returns or results while considering all factors that affect the environment. Optimisation is the general term that is used to describe the process of minimising or maximising objectives in this search of 'better'. Multi-objective optimisation (MOO) refers to the case where a problem has two or more conflicting objectives and these are optimised simultaneously while adhering to problem-specific constraints. This chapter is devoted to the field of MOO – explaining key concepts and providing an overview of the literature. With the field being as wide as it is, the chapter serves as the backdrop for the rest of the document and is restricted to paint a broad background and present elements of the field that will be used subsequently.

2.1 Theoretical foundations to multi-objective optimisation

The concept of multi-objective optimising developed naturally from single-objective optimisation. Researchers were faced with complex decision making and the need to develop a way to optimise two or more objectives simultaneously came to the fore. Kuhn & Tucker (1951) introduced the concept of the *vector maximum problem* and mathematically formulated a multi-objective problem (MOP), providing a foundation for the field in proceedings which is considered by Coello Coello *et al.* (2007) to be "the first serious attempt to derive a theory". It was these findings that enabled research to progress both theoretically and practically in the years that followed, establishing the field of multi-objective optimisation to what it is today (Sawaragi *et al.*, 1985). Although evolving from single-objective optimisation there are some key differences which differentiate the two fields. Due to the complexity of finding an optimum of more than one objective function, MOO is considered as a field in its own right with mathematical formulations, definitions and techniques. With this complexity in mind, Coello Coello *et al.* (2007) further explain that many MOPs are high-dimensional, discontinuous, multi-modal and/or NP-Complete, and solving these problems can be difficult.

Gil *et al.* (2007) provide the following definition:

> *Multi-objective optimisation is the process of searching for one or more decision variables that simultaneously satisfy all constraints, and optimise an objective function vector that maps the decision variables to two or more objectives.*

While a global optimum is not attainable in MOO, Coello Coello *et al.* (2007) define the unique term optimise as "the term of finding such a solution which would give the values of all the objective functions acceptable to the decision maker." Problems generally exhibit a set of solutions as opposed to the single-objective global optimum. The field is thus concerned with the representation of the trade-off of objectives in such a way that the decision maker can choose a set which is adequate in serving his/her needs. Originally introduced by Edgeworth in 1881 and generalised by Pareto in 1896 (Coello Coello, 2006), this set of solutions is defined as the Pareto optimum (originally the Edgeworth-Pareto optimum). The concept of this optimum and further definitions are discussed in the next section.

From the initial introduction in the 1950's, the field of multi-objective optimisation has grown steadily over the past decades, with a definite increase in published articles since the 1970's and a wide international research base as surveyed by Steuer *et al.* (1996). MOO has in fact enjoyed so much attention, that it has grown to encompass several smaller fields, especially characterised by the approaches followed to solve a MOP. The use of evolutionary algorithms for MOO singularly culminates to a formidable scholarship that is briefly discussed in section 2.3 along with other approaches to the problem.

Despite the amount of research available, the complexity of solving a MOP is constantly opening the door for even more research. With the foundations of the field firmly established, research has shifted toward improving set methods and solving new MOPs. Current research trends in the field of evolutionary MOO is reviewed by Coello Coello (2009) to counter questions being raised on the novelty of new research. From his review, certain topics are raised such as new algorithms, efficiency, relaxed forms of dominance, scalability and alternative metaheuristics; the latter placing emphasis on biologically-inspired metaheuristics. It is clear that the field is still vibrant and growing. In addition, most disciplines are faced with problems with such a set of conflicting objectives. Researchers are therefore constantly searching for valid approaches to these problems due to their complexity.

2.2 The multi-objective optimisation problem; definitions and concepts

This section provides an overview of the formulation of the multi-objective optimisation and explains and defines key concepts in the field. The formulation of the MOP is found in Coello Coello (2009) and documented below.

Definition 1: Decision variables: *The vector* $\mathbf{x} = [x_1, x_2, \ldots, x_D]^T$ *of variables for which numerical quantities are to be chosen in the optimisation problem.*

Restrictions are often imposed on an optimisation problem due to practical requirements, which must be satisfied for a solution to be acceptable. The constraints define the dependencies among decision variables and problem parameters (constants). The M inequality constraints are described by

$$g_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, M \tag{2.1}$$

and the Q equality constraints by

$$h_j(\mathbf{x}) = 0,\ j = 1, \ldots, Q. \tag{2.2}$$

The *degrees of freedom* are given by $M - Q$, and it is required that $Q < M$ to avoid an *overconstrained* problem.

This leads to the definition of a MOO problem with K objectives and $M + Q$ constraints in the case of minimisation,

$$\text{Minimise } \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_K(\mathbf{x})]^T \tag{2.3}$$

subject to

$$\mathbf{x} \in \Omega \tag{2.4a}$$
$$\Omega = \{\mathbf{x} \mid g_i(\mathbf{x}) \leq 0,\ i = 1, 2, \ldots, M; \tag{2.4b}$$
$$h_j(\mathbf{x}) = 0,\ j = 1, \ldots, Q\}. \tag{2.4c}$$

In single-objective optimisation a set of decision variables ($\mathbf{x} = [x_1, x_2, \ldots, x_D]^T$) is associated with one objective. A MOP is mapped in two Euclidian spaces, namely the decision variable space and the objective function space, where each vector of solutions in the first space is associated with a point in the latter. The formal definitions of the spaces are:

1. In the D-dimensional space each coordinate axis corresponds to a component of the vector $\mathbf{x}$.

2. In the M-dimensional space each coordinate axis corresponds to a component of the objective function vector $\mathbf{f}(\mathbf{x})$.

Figure 2.1 illustrates the concept of these two spaces in which optimisation is carried out. A typical algorithm will make decisions for optimisation based on the objective function space and consequently provide feasible solutions in the decision variable space that will return favourable results in the objective function space.

In single-objective optimisation there exists a global optimum. In the case of MOO the optimal solution is not clearly defined, but consists of a set of optimums, constituting the Pareto-optimal front (Gil *et al.*, 2007). Coello Coello *et al.* (2007) define the Pareto optimum as "the solution to a MOP if there exists no other feasible solution which would decrease some criterion without causing

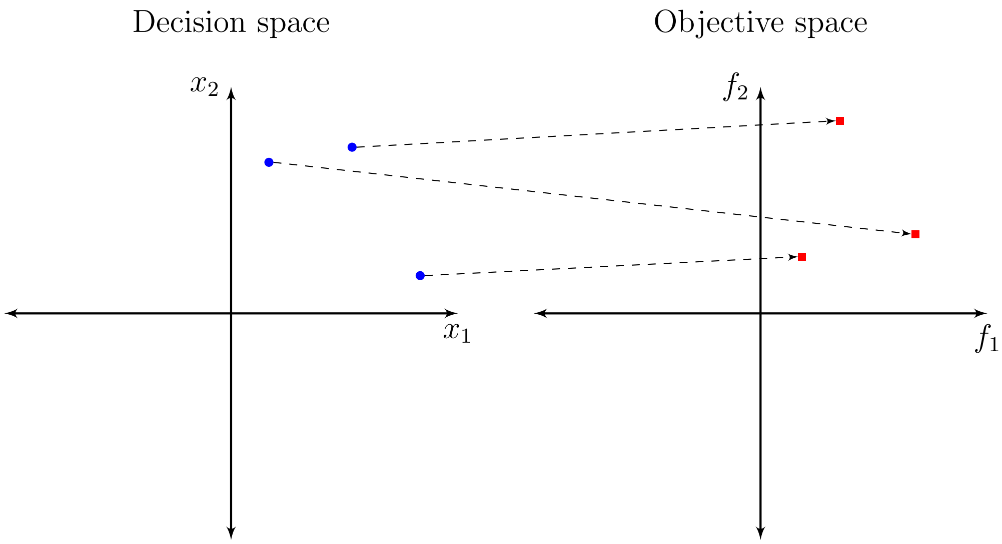

Figure 2.1: MOO euclidean spaces.

a simultaneous increase in at least one other criterion". This set of solutions is associated with vectors of decision variables which constitute the non-dominated set. MOO is concerned with finding this non-dominated set in as few evaluations as possible. proc A few definitions pertaining to Pareto optimality are necessary, and the basic definitions in Coello Coello (2009) are repeated here for convenience (assuming minimisation):

Definition 2: Given two vectors $\mathbf{u} = (u_1, \ldots, u_K)$ and $\mathbf{v} = (v_1, \ldots, v_K) \in \mathbb{R}^K$, then $\mathbf{u} \leq \mathbf{v}$ if $u_i \leq v_i$ for $i = 1, 2, \ldots, K$, and $\mathbf{u} < \mathbf{v}$ if $\mathbf{u} \leq \mathbf{v}$ and $\mathbf{u} \neq \mathbf{v}$.

Definition 3: Given two vectors $\mathbf{u}$ and $\mathbf{v}$ in $\mathbb{R}^K$, then $\mathbf{u}$ dominates $\mathbf{v}$ (denoted by $\mathbf{u} \prec \mathbf{v}$) if $\mathbf{u} < \mathbf{v}$.

Definition 4: A vector of decision variables $\mathbf{x}^* \in \Omega$ (Ω is the feasible region) is *Pareto optimal* if there does not exist another $\mathbf{x} \in \Omega$ such that $\mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{x}^*)$.

Definition 5: The *Pareto optimal set* $\mathcal{P}^*$ is defined by $\mathcal{P}^* = \{\mathbf{x} \in \Omega \mid \mathbf{x} = \mathbf{x}^*\}$.

Definition 6: The *Pareto front* $\mathcal{P}_T^*$ is defined by $\mathcal{P}_T^* = \{\mathbf{f}(\mathbf{x}) \in \mathbb{R}^K \mid \mathbf{x} \in \mathcal{P}^*\}$. The vectors in $\mathcal{P}^*$ are called *nondominated*, and there is no $\mathbf{x} \in \Omega$ such that $f(\mathbf{x})$ dominates $f(\mathbf{x}^*)$.

In some cases, determining the Pareto optimum set can be done visually, but this is not always possible. **Pareto dominance** (Definition 6) is the term used to define one set of solutions as being better than another (Goldberg, 1989).

Figure 2.2 illustrates the concept of non-dominated solutions in the objective function space, the Pareto-front.

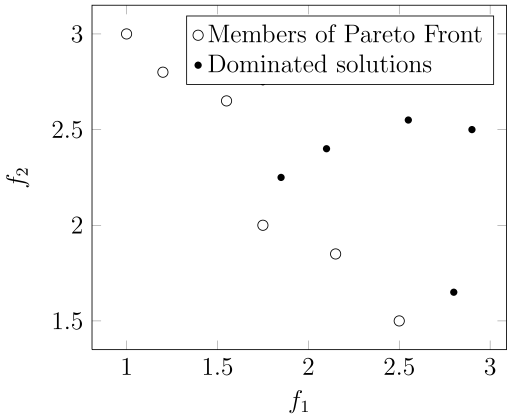

Figure 2.2: Illustration of the Pareto-front of non-dominated solutions (minimisation).

2.2.1 Ranking of solutions

The definitive aspect of MOO that distinguishes the field from single-objective optimisation is the need to **rank** solutions. In single-objective optimisation, a set of solutions are ranked from lowest to highest in the case of minimisation and from highest to lowest in the case of maximization; separating the good solutions from the rest of the population is intuitive. In contrast, MOO provides a challenge in ranking a population of solutions to enable the optimisation of a problem. In a comparative study of several ranking methods, Jaimes *et al.* (2009) show that different ranking methods produce different subsets of the Pareto optimal set and that the choice of method has a substantial influence on the quality of solution of any MOO method. They group ranking methods into those with or without parameters, favour ranking, preference order ranking and finally Pareto ranking. Fonseca *et al.* (1993) review the use of genetic algorithms in MOO and refer to the use of *Pareto-Based Approaches*, where selection (or reproduction in the case of GA's) is based on the dominance property of the objective values. A set of solutions is ranked according to the corresponding objective values and their relative position in the objective space. Goldberg (1989) for example uses a method where a rank 1 is given to non-dominated individuals in a certain population. These are removed and the individuals that exhibit non-dominance are assigned with rank 2. This is in contrast to the Pareto-based ranking method of Fonseca & Fleming (1995) who introduced a rank for each individual based on the number of individuals in a population that dominate it.

2.2.2 Performance measures

Due to the nature of estimated Pareto fronts, determining the performance of an algorithm is equally complex. Deb (2001) states the two goals of multi-objective optimisation as the convergence to the Pareto-optimal set and maintaining the diversity of solutions. Performance can therefore be directly measured by evaluating the attainment of these goals, and the quest for a single performance metric that incorporates these goals was launched. Some of the metrics suggested by literature and discussed in the tutorial by Fonseca *et al.* (2005) include the hypervolume indicator (Zitzler *et al.*, 2003), the unary epsilon indicator and the R_2 and R_3 indicators.

The hypervolume comparison method is a recognised unary indicator used in comparing two different Pareto-sets in order to assess the difference in quality of two algorithms. According to Zitzler *et al.* (2003) the hypervolume indicator (I_H) is the only unary indicator that is capable of detecting that a set of solutions is not worse than another. The hypervolume indicator does exhibit some weakness such as the large computational burden for a large number of objectives and the need to define a reference point. Due to the nature of the study this number of objectives never exceed two and consequently the use of the hypervolume (or hyperarea) indicator is supported.

The concept of the indicator is briefly explained in Figures 2.3 and 2.4, assume f_1 and f_2 must be minimised. It is difficult to distinguish between the two Pareto-sets illustrated in Figure 2.3 to decide on the best performance. A common reference point is selected that exceeds the maximum of both fronts on both axes, in the case of Figure 2.3 the reference point is set at $(5, 5)$. In Figure 2.4 the respective hyperareas are calculated and Front 1 returns a hyperarea of 10 and Front 2 a hyperarea of 8.6. Front 1 is subsequently selected as the better Pareto approximation.

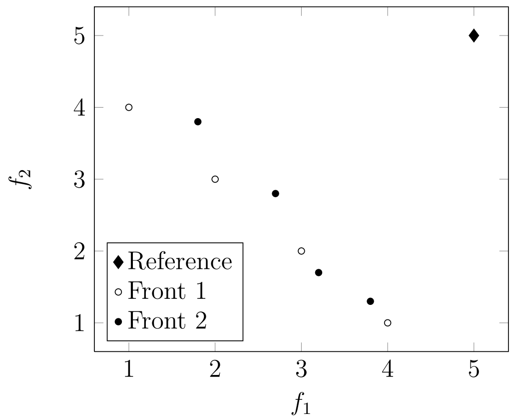

Figure 2.3: Illustration of the hyperarea indicator – determining the reference point.

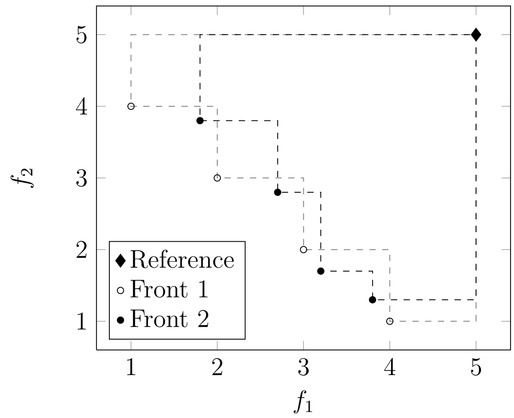

Figure 2.4: Illustration of the hyperarea indicator – calculating the areas.

2.3 Approaches to multi-objective optimisation

The field of multi-objective optimisation has been a research topic for a number of years, with various approaches being proposed throughout literature. Numerous deterministic optimisation approaches have been identified and successfully applied to a wide variety of problems. Coello Coello (2006) states that mathematical programming techniques are limited in solving MOPs due to a number of factors such as needing a differentiable objective function and/or constraint, generating only one solution per run and finally struggling to optimise problems with a disconnected or concave Pareto-front. These factors all contributed to the development of heuristic approaches to MOO. Multi-objective optimisation using evolutionary algorithms (MOEAs) is one of the major approaches found in the scholarly literature and actively researched. The two major references are the books by Deb (2001) and later Coello Coello *et al.* (2007), providing comprehensive literature on both the field of MOO and the use of evolutionary algorithms.

Jones *et al.* (2002) review the use of metaheuristics in the field of MOO with specific reference to genetic algorithms, simulated annealing and tabu search. A fourth primary approach is that of Monte Carlo methods as identified by Coello Coello *et al.* (2007). These approaches are briefly discussed in the remainder of the section.

The field of **evolutionary algorithms** (MOEA) is arguably the most popular, with 70% of the articles reviewed by Jones *et al.* (2002) utilizing genetic algorithms as their primary heuristic, possibly due to the general flexibility and diversity in possible approaches. Evolutionary algorithms are inspired by the "survival of the fittest" concept. Researchers are continually using search and optimisation techniques that are based on natural selection and genetics to solve a wide range of problems. The comprehensive reference of Coello Coello *et al.* (2007) serves as an encyclopaedic volume on the use of genetic and evolutionary computational algorithms for deriving the solution of MOPs. Evolutionary computing is the term for several stochastic search methods which computationally simulate the natural evolutionary process. This includes techniques such as genetic algorithms, evolution strategies and evolutionary programming. In a review of evolutionary methods at that stage, Fonseca & Fleming (1995) state that the evolutionary optimisation algorithm differs from conventional techniques as it is well-suited to multi-objective optimisation as opposed to other techniques that reformulate problems as single-objective. This is credited to the fact that multiple individuals search for multiple solutions in parallel. Other features of complex problems that are adequately handled by EAs include discontinuities, multi-modality, disjoint feasible spaces and noisy function evaluations (Fonseca & Fleming, 1995).

The Vector Evaluated Genetic Algorithm (VEGA) by Schaffer in 1985 was the first step toward using GAs in multi-objective optimisation. Building on this, the first Multi-objective Genetic Algorithms (MOGAs) were developed in 1993 (Fonseca *et al.*, 1993), who also introduced a rank-based fitness assignment method that allowed for intervention from the decision maker. In addition the complexity of large problems with a complex trade-off surface was dealt with by sampling small regions in a Pareto-based fashion, called niching.

Arguably the most famous GA is the Non-Dominated Sorting Genetic Algorithm (NSGA-II) by Deb *et al.* (2002). The NSGA-II is said to outperform other contemporary MOEAs such as the Pareto-archived evolution strategy (PAES) and the strength-Pareto EA (SPEA) in maintaining the diversity of solutions while converging close to a true Pareto-optimal set (Deb *et al.*, 2002). The properties of the NSGA-II include a fast non-dominated sorting procedure that decrease the computational burden of sorting the solutions into a non-dominated front. Other properties that contribute to the success of the NSGA-II are an elitist strategy, a parameterless approach and a simple yet efficient constraint-handling method (Deb *et al.*, 2002).

Other proposed GAs include the Niched Pareto Genetic Algorithm (NPGA) by Horn *et al.* (1994) which was one of the first MOO GAs documented in literature. The elitist MOEAs include the Strength Pareto EA by Zitzler & Thiele (1998) and the Pareto-archived Evolutionary (PAES) by Knowles & Corne (1999) and the elitist GA proposed by Rudolph (2001).

The past decade has seen the introduction of the Adaptive Range Multi-objective Genetic Algorithm (ARMOGA) by Sasaki & Obayashi (2005) that has achieved success in decreasing the computational burden for large scale problems with fewer evaluations.

The concept of **simulated annealing** as an iterative improvement heuristic to solve combinatorial optimisation problems was introduced by Metropolis *et al.* (1953) in the field of statistical mechanics with the analogy to optimisation drawn and later developed by a number of authors, including Kirkpatrick *et al.* (1983) and Hastings (1970). The method is analogous to the physical process of metallurgical annealing in which the metal is heated and cooled in a controlled way. The simulated annealing method for optimisation generates a random solution from a distribution of sample solutions which is accepted with probability depending on the difference between function values and the parameter T. It is particularly successful in avoiding a local optima by accepting solutions that lead to an increase in the function value as well as a decrease, as determined by means of the probabilistic acceptance criterion (Romeijn & Smith, 1994).

While often used in conjunction with other optimisation techniques, **tabu search** is a powerful method for solving combinatorial methods and escaping local optima. The method is based on two key elements, constricting the search by labelling certain moves as forbidden (tabu) and the clearing of the short term memory (Glover *et al.*, 1989). The method is often used in combination with a genetic algorithm, Glover *et al.* (1995) provide an initial reference on the similarities and possible formation of hybrid methods as a "marriage" between the two methods.

Monte Carlo methods are used in the optimisation of stochastic events which are simulated. The search in this case is purely random and independent of previous choices or outcomes (Coello Coello *et al.*, 2007).

The comprehensive list of available evolutionary algorithms and other methods shows the width of the field and in particular the great amount of multi-objective optimisation techniques available in literature. The chapter does not attempt to document the detailed working of every method but rather to show the wealth of the multi-objective optimisation field.

2.4 Concluding remarks on Chapter 2

As explained in the introduction, this chapter serves the purpose of providing a backdrop for the rest of the study. Multi-objective optimisation is a well established field with various approaches, research directions and problems. The chapter provided a brief overview of the literature on the subject, focusing on the development of the field, the main concepts and well-known methods. Elements such as definitions, ranking mechanisms, and performance measures set the scene for the rest of the document. The next chapter introduces and discusses the cross-entropy method.

CHAPTER 3

THE CROSS-ENTROPY METHOD

This chapter presents the outline for the rest of the study in the form of the cross-entropy method. The method is discussed by explaining the theoretical foundations, including the mathematical formulation and a review of the literature. This is then focused by a discussion of the theory regarding the application of the method to combinatorial optimisation and finally the application of the method to multi-objective problems is discussed as the focus of this study. The previous chapter painted the backdrop while this chapter chalks in the direction of the study.

3.1 Theoretical foundations to the cross-entropy method

Various methods are employed in the field of complex optimisation problems, culminating in a formidable scholarship of literature. Some of the well known stochastic methods for combinatorial optimisation were introduced in Chapter 2; simulated annealing, genetic algorithms and tabu search. The use of **randomised algorithms** in optimisation, *i.e.* methods that generate and use random variables in the optimisation of a problem, has gained in popularity over the past few decades. In general, these methods are favoured as the randomness ensures that the global optimum is more readily found. Some of the algorithms documented in literature are the stochastic comparison method introduced by Gong *et al.* (1992) and the nested partitioning method by Shi & Ólafsson (2000). A randomised method also showing successes as a multi-objective algorithm is the ant colony optimisation meta-heuristic by Dorigo *et al.* (1996). Another randomised method that falls into this category is that of the cross-entropy method (CEM), where an estimation problem is associated with the optimisation problem. The method is discussed in detail in the book of Rubinstein & Kroese (2004) and this section provides an overview of the method

The CEM was developed relatively recently by Rueven Rubinstein, based initially entirely on importance sampling. Rubinstein (1997) employed an algorithm in complex stochastic networks in computer simulation models to estimate the probability of a rare event. By assuming that the importance sampling distribution is a member of the same parametric family as the original distribution, he defined a two-stage procedure. In the first step the optimal parameter vector of the importance sampling distribution is estimated, and in the second step the optimal solution of the problem is estimated (Rubinstein, 1997). The basic concept was soon extended to solve combinatorial and continuous optimisation problems with a cross-entropy modification (Rubinstein, 1999).

The CEM for optimisation is based on two fundamental aspects, namely importance sampling and the Kullback-Leibler or cross-entropy distance. *Importance sampling* is generally used in Monte Carlo simulation as a variance reduction technique for increasing the efficiency of algorithms estimating integrals. Glynn & Iglehart (1989) summarise the technique as the replacement of the original random mechanism in a simulation by a new one while at the same time modifying the function being integrated (in the case of Monte Carlo algorithms). In the CEM, importance sampling is a variance reduction technique which refers to this iterative procedure where the system is simulated with a different probability distribution, or a different set of reference parameters, so as to make the occurrence of a rare event more likely (Rubinstein & Kroese, 2004). In the case of optimisation, this rare event refers to the obtainment of non-dominated objectives. A probability distribution that is problem-specific and related to the decision variables is used to sample a solution to the original problem.

While importance sampling is relatively well-known as a variance reduction technique, Rubinstein & Kroese (2004) feel that it has a drawback as the optimal reference parameters are often difficult to obtain with the conventional technique. In contrast the CEM provides a simple and fast adaptive procedure for estimating these optimal parameters. *Kullback-Leibler divergence* is a central index that is used in statistics as a qualitative measure of the similarity of two probability distributions (Lee & Park, 2006). For two probability distributions with density functions g and f, Lee & Park (2006) define the KL divergence of f from g as

$$\mathcal{D}(g, f) = \int g(x) log \frac{g(x)}{f(x)} \mathrm{d}x. \tag{3.1}$$

Optimisation is executed by minimising the value of (3.1) if f is the estimated probability distribution and g that of the optimisation problem, while employing importance sampling with the problem-specific probability distribution.

De Boer *et al.* (2005) summarise the CEM as follows:

- A data sample is generated according to a specified random mechanism.
- The parameters of the mechanism are updated according to the results obtained in the first step to ensure an improved sample.
- These steps are performed iteratively until a stopping criterion is met.

The first step is considered by many as the most important step in applying the method to various problems. A suitable probability family needs to be found that is well fitted to the structure of the problem (Margolin, 2005). The probability distribution should be set up to adequately sample feasible solutions to the problem. The original problem has to be associated with an estimation problem to employ the randomised method of CEM. As this estimation problem will vary for different problems, application of the method to a problem is largely founded in this first step. The last two steps are in general universal. The art of the application of the CEM for optimisation to two problems is mastered and discussed in **Chapter** 4 and **Chapter** 5. The next section presents the general application of the method to combinatorial optimisation problems (COPs).

3.2 The cross-entropy method for discrete combinatorial optimisation

Combinatorial optimisation refers to a problem where the decision maker seeks the correct combination of values of variables that will optimise the objective function. As soon as the different possible combinations of decision variables are investigated, the size of the decision space increases rapidly with an increase in decision variables. Due to the combinatorial nature of, for example the vehicle routing problem, it is clear that the computational complexity of the problem increases as the number of customers increase. In a similiar fashion, the case study presented in Chapter 5 has 194 decision variables that are combinatorial in nature. The CEM is presented as being able to solve such a COP. As mentioned previously, the main idea behind using the CEM for optimisation is to cast the COP on a rare-event estimation problem (ASP) which is then used in the algorithm. The outcome of this approach according to Rubinstein & Kroese (2004), is the "construction of a random sequence of solutions which converges probabilistically to the optimal or near-optimal solution of the COP". The attainment of said ASP is therefore the first step before the optimisation algorithm is applied to it and solutions generated are used to probabilistically determine the solution of the combinatorial problem.

The following theoretical discussion and formulation of the CEM for discrete combinatorial optimisation is based on Chapter 2 in the book by Rubinstein & Kroese (2004).

Let $\mathcal{X}$ be a finite set of states and f the performance or objective function on $\mathcal{X}$. For a general maximization problem, we seek the maximum of f over $\mathcal{X}$ and the states (decision variables) at this particular maximum (γ^*), *i.e.*,

$$f(x^*) = \gamma^* = \max_{x \in \mathcal{X}} f(x). \tag{3.2}$$

The CEM is set on simulating the rare event where $f(\mathbf{X})$ is larger than some real number γ. In order to associate an estimation problem with the general optimisation problem in (3.2), a collection of indicator functions on $\mathcal{X}$ is defined $I_{f(\mathbf{x})\geq\gamma}$ for various $\gamma \in \mathbb{R}$. A family of discrete probability densities on $\mathcal{X}$, parameterised by a parameter vector ($\mathbf{v}$), is also defined as $h(\cdot\,;\mathbf{v}), \mathbf{v} \in \mathcal{V}$. For a certain $\mathbf{u} \in \mathcal{V}$, the probability of estimating the event is expressed as

$$l(\gamma) = \mathbb{P}_{\mathbf{u}}(f(\mathbf{X}) \geq \gamma) = \mathbb{E}_u I_{f(\mathbf{X})\geq\gamma}.$$

Importance sampling was introduced in the previous section, and $l(\gamma)$ is estimated using this technique. A different density g to sample values from is defined on $\mathcal{X}$ and l is estimated by

$$\hat{l} = \frac{1}{N}\sum_{i=1}^{N} I_{f(\mathbf{x})\geq\gamma}\frac{h(\mathbf{X}_i;\mathbf{u})}{g(\mathbf{X}_i)}. \tag{3.3}$$

This estimation (3.3) is done by taking a random sample in $\mathcal{X}$ from a different probability function g. To enable importance sampling, g is chosen from the family of densities $h(\cdot\,;\mathbf{v})$, with $\mathbf{v}$ the reference parameter that will minimise the distance between density g^* and $h(\cdot\,;\mathbf{v})$. The Kullback-Leibler distance (3.1) is used and from this the associated stochastic problem is derived,

$$l(\gamma) = \mathbb{P}_{\mathbf{u}}(f(\mathbf{X}) \geq \gamma) = \sum_{x} I_{f(\mathbf{x})\geq\gamma} h(\mathbf{x};\mathbf{u}) = \mathbb{E}_{\mathbf{u}} I_{f(\mathbf{x})\geq\gamma}.$$

This estimation of parameters is done by defining a reference parameter $\mathbf{v}^*$,

$$\mathbf{v}^* = \underset{\mathbf{v}}{\operatorname{argmax}} \frac{1}{N}\sum_{i=1}^{N} I_{f(\mathbf{x})\geq\gamma} \ln h(\mathbf{X}_i;\mathbf{v}).$$

The idea is that if γ is close to γ^*, most of the probability mass of $h(\cdot\,;\mathbf{v}^*)$ will be assigned close to $\mathbf{x}^*$, and the distribution can be used to generate an approximate solution. Finally, in the case of discrete optimisation, the maximum of (3.2) can be derived by setting the first derivatives with respect to p_j equal to zero, which simplifies to

$$p_j = \frac{\sum_{i=1}^{N} I_{f(\mathbf{x})\geq\gamma} X_{ij}}{\sum_{i=1}^{N} I_{f(\mathbf{x})\geq\gamma}}. \tag{3.4}$$

For the purpose of the CEM algorithm, a smoothing update rule (with parameter α) is introduced in an effort to eliminate premature convergence, which is

$$\hat{\mathbf{p}}_t = \alpha\hat{\mathbf{p}}_t + (1-\alpha)\hat{\mathbf{p}}_{t-1}. \tag{3.5}$$

In the case of discrete combinatorial optimisation, an example algorithm is provided as Algorithm 1.

Algorithm 1 Example of the CE algorithm for discrete combinatorial optimisation (Rubinstein & Kroese, 2004)

1: Start with some $\hat{\mathbf{p}}_0$. Let $t = 1$.
2: Draw a sample $\mathbf{X}_1, \cdots, \mathbf{X}_N$ of vectors with probability vector $\hat{\mathbf{p}}_{t-1}$. Calculate performances $S(\mathbf{x}_i)$ for all i, and order them from smallest to biggest, $f_{(1)} \leq \cdots \leq f_{(N)}$. Let $\hat{\gamma}_t$ be sample $(1-\varrho)$-quantile of the performance.
3: Use this sample to calculate $\hat{\mathbf{p}}_t = (\hat{p}_{t,1}, \cdots, \hat{p}_{t,N})$ with (3.4).
4: Smooth $\hat{\mathbf{p}}_t$ vector with (3.5).
5: If the stopping criterion is met, **stop**; otherwise set $t \leftarrow t+1$ and return to Step 2.

On the topic of combinatorial optimisation, Rubinstein & Kroese (2004) devote a chapter to the travelling salesperson problem (TSP) and De Boer *et al.* (2005) also use the TSP as an example when illustrating the application of importance sampling to combinatorial optimisation. When comparing the CEM to other algorithms for combinatorial optimisation (simulated annealing, nested partitioning, tabu search, genetic algorithms), Rubinstein (1999) states that the CEM employs a global rather than a local search procedure, giving it a distinct advantage.

In the case of combinatorial optimisation, such as the travelling salesperson problem and vehicle routing problem, the probability distribution used in the importance sampling step is described by a transition probability matrix. A vehicle-specific matrix P is generated so that the probability of going to city j when in city i is represented as p_{ij}. The method starts out with equal probabilities and is then updated according to the routes in the current best solution set (elite) at each iteration with the following formula as explained by De Boer *et al.* (2005):

$$p_{t,j} = \frac{\sum_{i=1}^{N} I_{f(\mathbf{x}_i)\geq\gamma_t} I_{x_{ij}=1}}{\sum_{i=1}^{N} I_{f(\mathbf{x}_i)\geq\gamma_t}} \tag{3.6}$$

The probability of visiting city j (the jth success) is updated by (3.6) by counting the vectors with performance greater or equal than γ_t and with a j coordinate equal to 1 (j is visited). This is normalised by dividing by a count of all vectors with performance greater or equal than γ_t, *i.e.* all vectors in the elite set.

The worth of the method for optimisation is proved theoretically by Margolin

(2005) where two cross-entropy algorithms are tested for asymptotical convergence to the optimal solution. Although largely theoretical, this result can be compared to similiar tests on the ant colony optimisation method and simulated annealing.

3.3 Applications of the cross-entropy method for optimisation

As explained in the previous section, the CEM has been developed quite recently, but the favourable results and novel approach to optimisation led to a number of applications of the method. The CEM is presented as an efficient method for solving a variety of problems, including estimation, optimisation and Monte-Carlo simulation problems. The method is especially efficient in solving NP-hard combinatorial deterministic and stochastic problems as documented by Rubinstein & Kroese (2004).

Most of the applications of the method are documented in Rubinstein & Kroese (2004), and are generally confined to two broad categories, optimisation and rare-event simulation. The CEM is applied to optimisation problems once the underlying problem is translated into an associated estimation problem, or an associated stochastic problem (ASP) (De Boer *et al.*, 2005), that introduces randomness. The optimisation problem category is further grouped into deterministic and stochastic combinatorial problems. Some examples include the quadratic assignment problem, the maximal cut problem, clustering problems, the travelling salesman problem and the clique problem. In the case of stochastic or noisy problems, the method has been successfully applied to the buffer allocation problem, stochastic scheduling and flow control, to name a few.

Buffered service systems driven by Markovian fluid sources are often optimised with a simulation approach with a specific use of importance sampling. D'Acquisto & Naldi (2005) use a version of the CEM to overcome the problem of choosing the appropriate biasing conditions. The use of the CEM eliminates the need to determine the biasing conditions analytically before the start of the simulation, but conditions can be derived directly from the simulation (D'Acquisto & Naldi, 2005).

Ma (2011) successfully applied the CEM for single-objective optimisation to the vehicle routing problem with time windows. A multi-agent environment was introduced where a transition matrix is associated with each node of the network that is used to construct a feasible route for every vehicle (the agents in this case). It is this transition matrix that is constructed using a random mechanism and then updated according to the performance of the routes travelled by the vehicles, in effect increasing the probability of estimating the rare event of an optimum solution. The method proposed in Chapter 4 employs the same principle.

3.4 The cross-entropy method for multi-objective optimisation

Bekker & Aldrich (2010) adapted the cross-entropy method for multi-objective optimisation and tests on MOO benchmark problems exhibited favourable results. The core of the CEM was maintained but the multi-objective nature of the problem and solutions needed to be incorporated. In single-objective optimisation the elite solution used to update the probability distribution consists of the upper γ quantile which cannot be estimated in the case of multi-objective optimisation as the solution yields two or more objective values. A Pareto-ranking (Goldberg, 1989) was introduced to overcome this major difference, and the reader is referred to the original article by Bekker & Aldrich (2010) for a comprehensive explanation. The method, called MOO CEM, as adapted by Bekker & Aldrich (2010) showed satisfactory results as approximations of the true Pareto front was obtained with a relatively low number of simulations.

The CEM for MOO has been applied to a number of problems following the original article, which contained an application to a dynamic, stochastic adaptation of the well-known inventory problem and results of standard test problems (Bekker & Aldrich, 2010). A reconfigurable manufacturing system with optimisation objectives of minimising the work in progress and maximising the throughput was studied by Du Preez (2011) and the Pareto-front obtained by using the CEM for MOO. In the field of Process Engineering Bekker (2012) applied the method to an extrusion equipment design problem which is an deterministic, continuous MOO test problem developed by Gaspar-Cunha & Covas (2003). Finally Bekker (2012) also studied the application of the method to a stochastic combinatorial problem by applying the CE method to the widely researched buffer assignment problem as a multi-objective problem. In this well-known problem a finite number of spaces need to be allocated as buffer spaces between machines, while maximising the throughput rate and minimising the work in progress (WIP) in the MOO case. Bekker (2012) used discrete event simulation to model the problem with varying variables such as the number of resources or maximum allowable number of buffer spaces. It was found that the MOO CEM was efficient in finding good approximation fronts for various buffer allocation problems. The research showed a substantial decrease in the number of evaluations, contributing to a lower computational burden. It is this adaptation of the method as explained in this section that inspired this study and further applications are documented in the next chapters.

3.5 Concluding remarks on Chapter 3

The chapter provides a brief overview of the cross-entropy method for optimisation, examining its roots in the literature, explaining the fundamentals of the method and discussing the application of the method. The cross-entropy method has been proposed fairly recently as a tool for optimisation and has in its short lifetime exhibited favourable results in obtaining a global optimum without being too computationally intensive. The chapter concludes with a section on the introduction of the use of the method in multi-objective optimisation due to the successes obtained in single-objective optimisation and other advantages. It is this venture that forms the basis of the rest of the study. The following chapters document the application of the cross-entropy method for multi-objective optimisation to an academic discrete optimisation problem (the vehicle routing problem with soft time windows, chapter 4) and a real world discrete event simulation problem (blood inventory management in the Western Province, chapter 5).

CHAPTER 4

THE VEHICLE ROUTING PROBLEM WITH SOFT TIME WINDOWS

In the previous chapters the separate fields of multi-objective optimisation (MOO) and the cross-entropy method (CEM) were introduced. The field of MOO has grown in stature due to the value in the real world with evolutionary algorithms leading the research. The CEM is believed to be a successful global search procedure in optimising single-objective problems. A recent study has applied the CEM to MOO (Bekker & Aldrich, 2010). The previous chapter presented the method and the development as a multi-objective technique as the focus of the study. The research problem pertains to the application of this method in the field of combinatorial optimisation. This chapter presents the vehicle routing problem with time windows and the multi-objective optimisation thereof with the use of the cross-entropy method.

The concept of vehicle routing is evident in many operations of our daily lives. Products being delivered to a store, the routing of mail, the transportation of people and even the routing of electronic messages are all examples. In the current day and age of rising oil and fuel costs the minimisation of transportation costs remains an important objective of logistics and operations management. In fact, due to the soaring fuel prices, the transport segment of South Africa's logistics costs reached an all time high of R180 billion in 2010, 53.2 % of the logistics budget which amounts to 12.7% of the country's GDP (Viljoen, 2011).

The vehicle routing problem (VRP) remains one of the most studied problems in the field of Operations Research. It has many real world applications but exact methods require a considerable amount of computational time. In this chapter, the multi-objective version of the vehicle routing problem with time windows (VRPTW) is considered, with a focus on the VRP with soft time windows variant (VRPSTW). In this problem type, a number of vehicles have to provide a service to customers at different locations while adhering to constraints with regard to the capacity of the vehicle and the time window in which service should start. Although the problem has been considered as a multi-objective problem (MOP) by a number of authors, the focus has primarily been on minimising the number of vehicles and the total travel distances. This study considers these and other objectives with conflicting objectives. The aim of the chapter is to apply the MOO CEM discussed in the previous chapter to the VRPSTW.

The chapter is structured into the following sections: literature pertaining to the problem is discussed, followed with the formulation of the MOP model and the application of the optimisation method and concludes with a discussion of the results. The chapter is based on a submitted article (Hauman & Bekker, 2012).

4.1 Research design

The specific part of the research hypothesis that is addressed in this chapter can be defined as follows: the CEM for MOO can be used to solve the multi-objective VRPSTW. This hypothesis was formulated after an extensive literature study on the field of the multi-objective VRPTW that is documented in the next section. The objectives of this chapter are listed below:

- Master the scholarship in the field of the VRPTW with a specific focus on MOO.
- Mathematically model the problem under consideration.
- Master the use of the CEM in combinatorial optimisation.
- Integrate the multi-objective cross-entropy algorithm with the VRPSTW problem.

- Use Matlab® to implement the cross-entropy algorithm and the VRPSTW route construction algorithm.
- Identify suitable benchmark problems.
- Apply the optimisation model to benchmark problems.
- Document and interpret results.

4.2 Literature review of the multi-objective vehicle routing problem with time windows

The literature review is constrained to literature applicable to the study and is accordingly divided into two sections, the first provides an overview of the origins of the problem with a specific focus on the time windows variation. The second section reviews the work done in MOO on the VRP.

4.2.1 The vehicle routing problem with time windows

In logistics and distribution, decision makers are often faced with the problem of developing optimal routes for vehicles that service different customers. The vehicle routing problem is considered to be a variation of the travelling salesperson problem where one salesperson has to visit a certain number of cities before returning to the home city. Also termed the "truck dispatching problem", Dantzig & Ramser (1959) considered it a generalisation of the travelling salesperson problem (TSP). In the vehicle routing problem, a number of vehicles needs to be routed to geographically dispersed nodes or customers. In addition, vehicles have limited capacity, which places a restriction on the number of nodes that one vehicle can visit. The vehicles also perform a service at the different nodes.

VRP models are widely studied and are in general involved with the distribution of goods with typical activities including pick-up and delivery. According to Crainic & Laporte (1997) the essence of the VRP is comprised in the design of pick-up and delivery routes. Due to the width of the application field the VRP has evolved into different sub-problems.

In its simplest form, the classical VRP entails finding a set of routes at minimum cost for a set fleet of vehicles with each vehicle starting and ending at the depot, and ensuring each vertex (customer) is visited by a vehicle. From this classical problem the field has been extended to comprise several problem classes with additional constraints and variations.

When the vehicles have a restricted capacity for goods to be delivered from the depot to the customers, the problem is defined as a capacitated vehicle routing problem (CVRP) and when there is a restriction on the distance travelled by each vehicle it gives rise to a distance constrained VRP (DVRP) (Toth & Vigo, 2002). Another variation is the multiple-depot VRP where a company has more than one depot from where service is provided to the customers (Giosa *et al.*, 2002) and the VRP with pick-ups and deliveries (VRPD). The latter is a variation of the classic problem with the absence of a normal depot but with multiple depots and service that is dependent on a pick-up and delivery point and the volume to be transported (Toth & Vigo, 2002).

To enable more realistic representations of real world problems the classical problem is extended to the stochastic VRP and finally the VRP with time windows (VRPTW). The VRPTW has many applications, such as the routing of buses and trains, bank deliveries and postal deliveries.

The problem under consideration in this study is the vehicle routing problem with time windows. In this problem, the time in which a vehicle may arrive to begin service at a certain node is limited to a predefined time window. The VRP with soft time windows further implies that vehicles can arrive after the time window has closed, although this is often associated with a penalty cost for late arrival. In the case of hard time windows no delivery is allowed after the close of a time window and the solution is regarded as infeasible.

In the VRPTW, a set of vehicles with limited capacity is to be routed from a central depot to a set of geographically dispersed customers with known demands and predefined time windows (Tan *et al.*, 2006).

Toth & Vigo (2002) summarise the concept of the VRPTW, as applicable to this article, as follows:

1. Each route visits the depot vertex.

2. Each customer vertex is visited by exactly one route (within the specified time window).

3. The total demand of the customers visited by a route do not exceed the capacity of the separate vehicles.

The addition of time windows increases the complexity and computational intensity of the problem. The VRP is classified as a NP-hard problem and subsequently the VRPTW is a constrained NP-hard problem. Savelsbergh (1991) showed that even finding a feasible solution for the VRPTW with a fixed number of vehicles in the fleet is a NP-complete problem. Ioannou *et al.* (2003) explain that the use of hard time windows often place an upper constraint on the possible solutions as meeting these time windows can lead to an increase in the fleet size of the company. By creating a MOP in which the fleet size and the time window violations are minimised, the soft time window problem provides better fleet size results with minimal time window violations. Taillard *et al.* (1997) claim that in the relaxation of soft time windows feasible solutions are easier to find as there are fewer hard constraints, but further state that this is countered by the way hard time windows in turn allow for infeasible solutions to be filtered out fairly quickly. Finding a solution technique that is computationally effective is therefore often primarily concerned with heuristic algorithms.

The VRP in all its forms is studied at the hand of various solution techniques, with algorithms usually divided into three categories: exact methods, heuristics and metaheuristic methods.

As explained earlier the definition of the VRPTW as a NP-hard problem supports the need for heuristic algorithms in which a large solution space is traversed by an experience-based search rather than an exhaustive search. Heuristics are generally used as route improvement algorithms and considered to be computationally efficient. The paper by Solomon (1987) on various algorithms in the case of the VRPTW is widely cited. In this paper a variety of tour-building heuristics are described and studied in terms of their performance. These heuristics are briefly discussed here. The first is the *savings heuristic* that is adapted from the original by Clarke & Wright (1964) by investigating links between two end customers which will result in the smallest increase in cost (distance).

The second heuristic considered by Solomon (1987), is a *time-oriented, nearest neighbour heuristic* which is a sequential, tour-building algorithm that adds the closest unrouted customer to the route. The closest customer is determined by a metric that accounts for both the geographical distance, the available time and the urgency of delivery. A generalization of this heuristic is the *insertion heuristic* in which the criteria determines which customer can be inserted between customers on an existing partial route. Lastly a *time-oriented sweep heuristic* is introduced in which the VRPTW is split into a clustering stage and a scheduling stage. During the first stage a sweep heuristic is used to group customers into vehicle-specific groups. The second stage entails the scheduling/routing of the vehicle to the selected customers (Solomon, 1987).

Metaheuristics are used to optimise a problem by iteratively improving a candidate solution. VRP solutions that fall into this category include ant colony optimisation, genetic algorithms, simulated annealing and tabu search. Tan *et al.* (2001) investigated various heuristic methods for the VRPTW including simulated annealing (with an updated cooling scheme), tabu search and a genetic algorithm implementation with integer string representation for the candidate routes, new crossover operations, hill-climbing and mutation schemes. Initially, the tabu search algorithm showed favourable results on the constrained VRP, with Gendreau *et al.* (1994), Taillard *et al.* (1997) and Cordeau *et al.* (2001) publishing sucessful studies. The genetic algorithm and its variants have in turn gained popularity in the last decade. Berger & Barkaoui (2003) proposed a hybrid genetic algorithm for the capacitated VRP. In the case of the VRPTW, Bräysy & Gendreau (2005) present a survey on the research in the field at that point in time. The heuristics are grouped into route construction heuristics and solution improvement heuristics, with the first part largely influenced by the contribution of Solomon (1987).

As explained in Chapter 2 metaheuristics are particularly useful in MOO. As the focus of this study is the MOO of the VRPTW the next section is devoted to the field with a specific focus on the metaheuristics that are used.

4.2.2 Multi-objective optimisation and the VRP

In this section, the work of a few researchers in the field of multi-objective optimisation (MOO) and the VRP is mentioned, followed by a discussion of benchmark problems in the research field.

The VRP is generally viewed as having the single objective of minimising the distance travelled by the different vehicles. However, over the past few years the problem has been considered as being multi-objective, with objectives including the distance travelled, average lateness and the number of vehicles. Jozefowiez *et al.* (2008) review the field of multi-objective VRPs. They identify three uses of multi-objective VRPs: an extension of classic academic problems in order to improve their practical applications, general classic problems and the study of real-life cases where the decision maker identified the objectives.

Jozefowiez *et al.* (2008) further group the methods used to solve the MOPs into scalar methods, Pareto methods and a third category that considers different objectives separately. While it is possible to generate a weighted cost function with regard to two of the objectives in order to use scalar methods, this would generally result in a bias towards one of the objectives. The field of MOO has been developed to display the trade-off between objectives in an objective way.

When considering the variation of the VRP with time windows, it appears that in the field of Pareto methods, evolutionary algorithms are used in most cases. Ghoseiri & Ghannadpour (2010) model the problem with a goal programming approach, minimising the undesired deviations from the aspiration level of the objectives. From this model solutions are found by a genetic algorithm and the Pareto ranking scheme is used to find the non-dominated solutions. Tan *et al.* (2006) propose a hybrid multi-objective evolutionary algorithm (MOEA) and Ombuki *et al.* (2006) use a multi-objective genetic algorithm.

Geiger (2008) states that the relaxation of the time window restriction (VRPSTW) allows for a more practical multi-objective formulation and investigates the influence of this relaxation and other problem characteristics on genetic operators in evolutionary algorithms. Garcia-Najera & Bullinaria (2011) recently proposed an improved MOEA which uses a similarity measurement to enhance the diversity of solutions. When compared to general evolutionary methods such as the popular NSGA-II (Deb *et al.*, 2002), this method shows improvements, particularly in preserving high diversity before settling on a solution. Garcia-Najera & Bullinaria (2011) further demonstrated that 26 instances of Solomon's benchmark set exhibit conflicting objectives, and emphasised the use of a Pareto front. While the use of evolutionary algorithms is valid, this study investigates the use of the CEM as an alternative approach to solve the MOO VRPSTW.

In the case of benchmark problems it was found that, in addition to providing algorithms for solving the VRPTW, Solomon (1987) also developed six sets of benchmark problems that have since been used for comparing different methods. Although the Solomon benchmark set has been extended and used in most literature on multi-objective vehicle routing, Castro-Gutierrez *et al.* (2011) found that classic test instances such as these problems developed by Solomon are not entirely suitable for MOO. The objectives used by Garcia-Najera & Bullinaria (2011) (the number of vehicles and the total travel distance) were in fact found to be in harmony for most of Solomon's problems. (This finding was supported when tests on these objectives with the method proposed in this study exhibited a Pareto front with one or at most two points.) Alvarenga *et al.* (2007) state that the objectives (number of vehicles and the total travelling distance) represent concurrent objectives.

The need for specific multi-objective test cases as opposed to the extension of traditional single-objective cases came to light and a set of problem instances were generated by Castro-Gutierrez *et al.* (2011) to address this need. Initial experiments with the evolutionary algorithm NSGA-II (Deb *et al.*, 2002) showed evidence of multi-objective features such as a correlation value close to -1 or 1 for a given pair of objective values. In the case of minimisation, a correlation value close to -1 indicates a pair of objectives with a conflicting nature, *i.e.*, the minimisation of one objective leads to an increase in the other objective. Due to these findings, the proposed CEM for MOO was applied to the newly proposed benchmark problems. Some reference solution sets for researchers wishing to do further work in MOO of the VRPSTW is also presented and can be found in Appendix A. Castro-Gutierrez *et al.* (2011) identified five objectives to be used in vehicle routing (see Table 4.1): the number of vehicles ($Z1$), the total travel distance ($Z2$), the makespan (travel time of longest route) ($Z3$), the total waiting time when vehicles arrive before the time window ($Z4$) and the total delay time when vehicles arrive after the time window ($Z5$).

To apply the CEM for MOO two critical steps are to define an estimation problem as explained in Chapter 3 and to isolate a Pareto-front of solutions as needed for MOO. The expression in (3.6) is applicable to the singe-objective optimisation as applied by Rubinstein & Kroese (2004) and De Boer *et al.* (2005). The elite is defined as instances with an objective function value ($f(\mathbf{x}_i)$) smaller than γ. The ranking of the elite in the case of MOO as proposed by Bekker & Aldrich (2010) and the subsequent construction of the probability matrix (to model the estimation problem) as used in this study are explained in Section 4.3 with the formulation of the optimisation model.

Section 3.3 mentioned the work by Ma (2011), in which the CEM for single-objective optimisation to the VRPTW was successfully applied. A multi-agent environment was introduced where a vehicle-specific transition matrix is associated with each node of the network that is used to construct a feasible route for every vehicle (the agents in this case). As explained previously, it is this transition matrix that is constructed using a random mechanism. It is then updated according to the performance of the routes travelled by the vehicles, in effect increasing the probability of estimating the rare event of an optimum solution. The proposed method employs the same principle, but uses a general transition matrix to construct routes for all the vehicles.

Ma (2011) further proposed a local search procedure on a subset of good solutions to avoid premature convergence by the CEM. The results on classic MOO test-instances by Bekker & Aldrich (2010) suggest that this is not necessary, but further research can investigate the influence of a local search procedure on the proposed multi-objective method in vehicle routing.

4.3 Formulation of the VRPSTW model

The VRPSTW consists of a network of customers at different locations indicated by coordinates, distances and travel times between customers and a central depot from where a homogeneous fleet of vehicles depart. The model used to achieve the research aim consists of a problem and optimisation method formulation and is presented in this section. The final solution consists of a list of the routes travelled by the different vehicles, usually illustrated by the order in which cities are visited. The problem was modelled with the following data structures: customer detail, distance and travel time between pairs of customers, and the proposed routes (candidate solutions).

4.3.1 Problem model of the VRPSTW

Kallehauge *et al.* (2005) define the VRPTW in mathematical terms with a fleet of vehicles, $\mathcal{V}$, a set of customers $\mathcal{C}$ and a directed graph $\mathcal{G}$. $\mathcal{N}$ is the set of vertices, $0, 1, ..., n+1$ with 0 and $n+1$ representing the depot (respectively the starting and returning depot). Define x_{ijk} as

$$x_{ijk} = \begin{cases} 1, & \text{if vehicle } k \text{ drives directly from vertex } i \text{ to vertex } j, \\ 0, & \text{otherwise.} \end{cases} \tag{4.1}$$

Further definitions are the capacity of each vehicle (Q), the demand of each customer i (D_i), the cost (or distance) d_{ij} and time (t_{ij}) associated with each arc (i, j) where $i \neq j$. The time window $[a_i, b_i]$ is associated with customer i. In the case of hard time windows the vehicle must arrive at customer i before b_i and in the case of soft time windows a delay time is logged. If a vehicle arrives before the time window starts, it incurs a waiting time until a_i when the service can start. The variable s_{ik} denotes the time at which vehicle k starts service at customer i. This is defined for every vehicle k and customer i, but becomes irrelevant if vehicle k does not service customer i. It is assumed that the time window of the depot always starts at zero and no service is required, *i.e.*, $s_{0k} = 0$, and the time back at the depot (although no service is required) is defined as $s_{(n+1)k}$.

For the purpose of this study, the mathematical model of Kallehauge *et al.* (2005) was adapted for the multi-objective VRP with soft time windows as shown in (4.2a) to (4.2e).

The optimisation model is:

$$\text{Minimise } k \tag{4.2a}$$

$$\text{Minimise } \sum_{k \in \mathcal{V}} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}} d_{ij} x_{ijk} \tag{4.2b}$$

$$\text{Minimise } \left(\max_k \left(s_{(n+1)k} \right) \right) \tag{4.2c}$$

$$\text{Minimise } \sum_{k\in\mathcal{V}} t_w^k \tag{4.2d}$$

$$\text{Minimise } \sum_{k\in\mathcal{V}} t_d^k \tag{4.2e}$$

subject to

$$\sum_{k\in\mathcal{V}}\sum_{j\in\mathcal{N}} x_{ijk} = 1 \qquad \forall i \in \mathcal{C}, \tag{4.3a}$$

$$\sum_{i\in\mathcal{C}} D_i \sum_{j\in\mathcal{N}} x_{ijk} \leq q \qquad \forall k \in \mathcal{V}, \tag{4.3b}$$

$$\sum_{j\in\mathcal{N}} x_{0jk} = 1 \qquad \forall k \in \mathcal{V}, \tag{4.3c}$$

$$\sum_{i\in\mathcal{N}} x_{ihk} - \sum_{j\in\mathcal{N}} x_{hjk} = 0 \qquad \forall h \in \mathcal{C}, \forall k \in \mathcal{V}, \tag{4.3d}$$

$$\sum_{i\in\mathcal{N}} x_{i,n+1,k} = 1 \qquad \forall k \in \mathcal{V}, \tag{4.3e}$$

$$x_{ijk}(s_{ik} + t_{ij} - s_{jk}) \leq 0 \qquad \forall i, j \in \mathcal{N}, \forall k \in \mathcal{V}, \tag{4.3f}$$

$$t_w^k = \sum_{i\in s_{ik}<a_i} (a_i - s_{ik}) \qquad \forall k \in \mathcal{V}, \tag{4.3g}$$

$$t_d^k = \sum_{i\in s_{ik}>b_i} (s_{ik} - b_i) \qquad \forall k \in \mathcal{V}, \tag{4.3h}$$

$$x_{ijk} \in \{0, 1\} \qquad \forall i, j \in \mathcal{N}, \forall k \in \mathcal{V}. \tag{4.3i}$$

Five conflicting objectives are defined in (4.2a) to (4.2e) and optimised in pairs. The five objectives and corresponding expressions are shown in Table 4.1 with the labels defined by Castro-Gutierrez *et al.* (2011).

The model differs from that of Kallehauge *et al.* (2005) in that the adaptation for soft time windows in (4.3g) calculates the total time a vehicle waits for a time window to start on a route (t_w^k), and (4.3h) calculates the total delay time of customers on a route waiting for vehicles that arrive after the close of a time window, denoted by t_d^k. The other constraints follow the original model. Constraint (4.3a) ensures each customer is visited exactly once, (4.3b) is the

Objective	Label	Expression
Number of vehicles	Z1	(4.2a)
Total travel distance	Z2	(4.2b)
Makespan of tasks	Z3	(4.2c)
Total vehicle waiting time	Z4	(4.2d)
Total vehicle delay time	Z5	(4.2e)

Table 4.1: Objectives of the VRPSTW (Castro-Gutierrez *et al.*, 2011).

capacity constraint and (4.3i) the integrality constraint. Constraints (4.3c), (4.3d) and (4.3e) ensure that each vehicle leaves the depot, arrives at a customer and then proceeds to the next customer, and that all vehicles end at the depot. Constraint (4.3f) incorporates the travel time between two customers. This model with the soft time windows will be used as platform for further analysis.

4.3.2 Optimisation model of the VRPSTW

The pseudo-code for the MOO CEM for solving the VRPSTW is shown in Algorithm 2, and the structure of the optimisation model is shown in Figure 4.1. An initial probability matrix (P) with equal probabilities is defined to start the process. The probabilities are then used to construct routes in order to obtain a set of N solutions, with q the identifier (index) of the particular set of routes. This construction of routes and normalisation of the probability matrix is summarised in Algorithm 3. The solutions obtained in this manner are then evaluated and values are assigned to the pair of objectives being studied (in Figure 4.1, Z_1 and Z_2 are used). The solutions are ranked using the algorithm of Goldberg (1989), and the ranking value is stored in ρ_q. The degree of domination of one vector relative to the other vectors in the solution set is indicated by this value. All vectors with $\rho_q \leq t_h$ after ranking of an iteration are weakly dominated and form part of the elite set of that iteration. The value of t_h is a preset threshold value, typically $0 \leq t_h \leq 2$. If $t_h > 0$, then weakly and non-dominated vectors are returned in the elite set. If $t_h = 0$, then only non-dominated vectors are added to the elite set. This elite set of solutions is used to obtain the updated probability matrix as defined by (3.6) and explained in Section 3.2. The expression in (3.6) is adapted to allow for the multi-objective case shown by,

$$p_{ij} = \frac{\sum_{q=1}^{N} I_{\{\rho_q=0\}} I_{\{x_{ij}=1\}}}{\sum_{q=1}^{N} I_{\{\rho_q=0\}}}. \tag{4.4}$$

Indicator values for every solution (q) with a number of routes (k) are summed in the population (N) as opposed to the TSP (see (3.6)) where a solution consisted of one tour. This process is iterated until the stopping criterion is met.

The expression in (4.4) can be explained with reference to Figure 4.1. Suppose the solutions with indices 1, 2 and N end up as the three solutions of the elite. The denominator will be equal to 3, while the various x_{ij} must be considered in the numerator. It can be seen that x_{09} occurs in all three solutions, so $p_{09} = 3/3 = 1$, but p_{02} occurs in two of the three solutions, so $p_{02} = 2/3$.

The algorithm applied to the vehicle routing problem with time windows is illustrated in Algorithm 2, and is based on the work of Bekker & Aldrich (2010).

Algorithm 2 MOO VRPSTW with the CEM

```
Let P be the transition matrix of probabilities, N_m the maximum number
of loops, τ the maximum number of evaluations per loop and μ the set of
averages of the objectives in the elite array.
L_c ← 0
repeat
    t ← 0
    Initialise P_t and μ_t
    repeat
        t ← t + 1
        Construct routes using Algorithm 3 and P_t
        Evaluate routes
        Rank the solutions using the threshold t_h = 2
        Update P_t using (3.6)
        Smooth P_t ← αP_t + (1 − α)P_{t−1}
    until |μ_t − μ_{t−1}| ≤ ε or t ≥ τ
    Rank with t_h = 1
    L_c ← L_c + 1
until L_c > N_m
Rank with t_h = 0
Return elite set
```

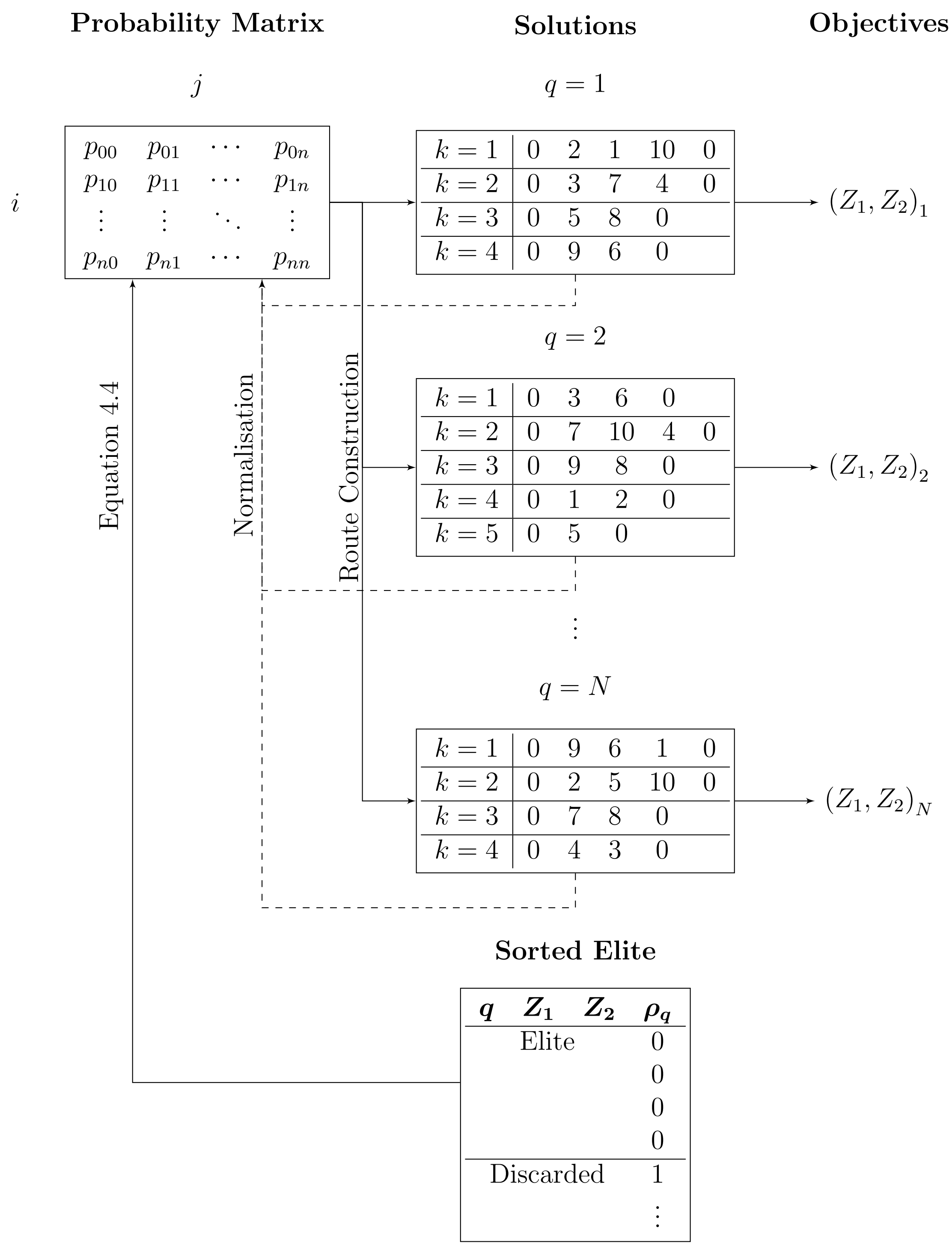

Figure 4.1: The structure of the optimisation model.

For the purpose of Algorithm 2 it is important to note that a rank of 0 denotes non-domination and accordingly the best solutions of the current set. The loop of steps 6 to 13 retains solutions with a rank of 2 to maintain solution diversity and to prohibit premature convergence. The non-dominated solutions are only isolated at the end of the algorithm, in step 17.

The values N_m (the maximum number of loops) and τ (the maximum number of evaluations per loop) are determined by the decision maker. N_m determines the number of times P_t and μ are initialised, in effect ensuring the diversity of the algorithm, while τ determines the number of times P_t is updated and solutions are sampled, ensuring the quality of solutions. The biggest consideration in limiting these values is on the time available. In this study for example, values were set to $N_m = 10$ and $\tau = 25$ which were found to be adequate in ensuring progression to an approximation front.

In principle, routes are constructed with Markov-chain transition probabilities but in reality the VRP is highly constrained and routes are dependent on the feasibility of said transition. Algorithm 3 ensures that the transition to non-feasible cities is not possible by setting the transition probability to 0 and normalising the remaining probabilities. The matrix with transition probabilities is updated according to feasible cities to be visited. From the array of feasible cities the next city to be visited is decided at random.

Algorithm 3 Construction of routes of the VRPSTW

```
Let depot be customer i + 1 (next customer)
while all customers are not visited do
    Previous customer i = Customer i + 1
    for all customers not yet visited do
        feas_1 ← (capacity of truck)
        feas_2 ← (time window (in the case of hard time windows))
        feas_3 ← (cut-off time back at the depot)
        Feasible ← feas_1 and feas_2 and feas_3
        if Not Feasible then
            Change probability to be visited from current customer to 0
        end if
    end for
    if No customers are feasible then
        The next customer is the depot.
        Routes k ← k + 1 return to 2.
    else
        Calculate the new weighted row of the probability matrix.
        Sample a customer from this row.
        Add this customer to the route as customer (i + 1)
    end if
end while
```

4.4 Methods and results

The algorithm for the CEM was coded in Matlab® and applied to the different benchmark instances of Castro-Gutierrez *et al.* (2011).

4.4.1 Benchmark problems

Castro-Gutierrez *et al.* (2011) generated a benchmark problem set that can be used for the multi-objective VRPSTW. The time windows and the demands of the customers are characterised in a certain way. The problems are characterised with the number of customers, the different time window profiles (0–4, 4 being the tightest) and capacity constraints (0–2, 2 being the tightest). *I.e.*, time window profile 0 would typically entail a time window of 288 000 time units while an example time window profile 4 has an average time window of 9 506 time units. Capacity profile 0 for the 50 customer problem means vehicles have a capacity of 690 units while capacity profile 2 for the 50 customer problem implies vehicles with a capacity of 85. Each problem has a specific label, for example, "50_d0_tw1" denotes the benchmark problem with 50 customers, a demand/capacity profile number 0 and time window profile 1. Six pairs of objectives (from the five defined in Table 4.1) were found to be conflicting in the datasets for soft time windows, namely $(Z1, Z3), (Z1, Z5), (Z2, Z3), (Z2, Z5), (Z4, Z3)$ and $(Z4, Z5)$ (Castro-Gutierrez *et al.*, 2011). These objectives are used in the testing of the CEM for MOO. Instances with negative correlation values (close to -1) for all pairs of objectives were used, *i.e.*, for all demand/capacity profiles and for the time window profiles (1–4) as the time window profile 0 only exhibits a time window for the depot and the problem reduces to a general VRP (no waiting time or delay time).

4.4.2 Performance measures

The hypervolume comparison method yields a recognised unary indicator used in comparing two different Pareto sets in order to assess the difference in quality of two algorithms. The hypervolume indicator (I_H) is also the only unary indicator capable of detecting that a set of solutions is not worse than another (Raad *et al.*, 2011; Zitzler *et al.*, 2003). This indicator is used to isolate the best result of 10 pseudo-independent tests of the MOO CEM for the VRP. To show the general performance of the method, the highest, lowest and average values for I_H are documented for the 10 runs. Since this study deals with pairs of objectives, we shall further refer to the hyper*area* indicator instead of the hypervolume indicator.

The indicator values of all problem sets on which the algorithm was tested are shown in Table 4.2 and Table 4.3. It is important to note that due to the difference in the order of the objectives, the hyperarea is a relational indicator, *i.e.*, a high value in one column is incomparable to the seemingly lower value in another column. It is also interesting to note that in some cases (especially in the $Z1$ and $Z5$ pairing) the hyperarea indicator is equal to 0 for all the documented indicator values of particular problem cases. This shows that the algorithm performed the same for all 10 runs as there was no difference in the hyperarea of the final approximate Pareto fronts. This is applicable to discrete cases such as the number of vehicles ($Z1$) and in the case of $Z4$ and $Z5$ when an optimal solution (such as delay time $= 0$ or waiting time $= 0$) is found in all 10 iterations. In these cases the movement of the second objective is irrelevant as the hyperarea indicator depends on an area, and here the extreme point of this objective forms a line with the secondary objective, thus resulting in an indicator value of 0. Variances higher than 0.5 are indicative of answers that did not exhibit a final multi-objective front (*i.e.*, one optimal point) but the "best" iteration provided a solution better than the other iterations, resulting in a large variation in the hyperarea indicator value.

4.4.3 Parameter setting

As the aim of the research is to determine if the MOO CEM can in general be used for the VRPSTW, the fine-tuning of parameters is not investigated. However, before tests were performed on the benchmark problems, experiments were conducted to get an indication of good parameter values. The main parameters that influence results include the population size N, the cross-entropy smoothing parameter value of α, and lastly the maximum allowed number of iterations (τ and N_m), as explained in the previous section.

A number of experimental tests were conducted and the averages of the I_H indicator of five runs were computed from a common maximum reference, $f_1 = 2\,138$ and $f_2 = 28\,800$. In Figure 4.2 the population size was set at $N = 2\,000$ and the average I_H of five runs at different values of α showed $\alpha = 0.9$ performed the best. In Figure 4.3, the value of $\alpha = 0.8$ was set while different values of N were tested. The difference in the indicator values for $N = 2\,000$ and $N = 2\,500$ was deemed not large enough to warrant the increase in computational time associated with $N = 2\,500$. Following these experiments and due to time considerations, tests were conducted with parameters set to $N = 2\,000$, $\alpha = 0.9$, $\tau = 25$ and $N_m = 10$.

4.4.4 Results of the VRPSTW

Tests were performed on all cases of the 50 customer problems with conflicting objectives, but only one problem instance is discussed in detail in this section, namely 50_d1_tw4. This problem instance is chosen as a fair representation of results found across the different demand and time windows, and serves as a good platform to discuss the performance of the algorithm and the findings of the study.

		Z1-Z3	Z1-Z5	Z2-Z3	Z2-Z5	Z4-Z3	Z4-Z5
50_d0_tw1	best	2 160	25 380	413 334.0	3 857 736	69 721 200	0
	average	1 704	4 716	375 580.8	2 645 854	62 858 160	0
	worst	1 380	0	343 026.0	1 641 396	53 330 400	0
50_d0_tw2	best	3 120	0	287 310.0	1 030 632	88 077 600	291 600
	average	2 964	0	258 177.0	621 264.6	80 239 680	150 120
	worst	2 820	0	230 874.0	63 354	73 411 200	0
50_d0_tw3	best	300	0	477 456.0	21 774 401	11 361 600	2.21E+08
	average	258	0	402 969.6	19 475 492	9 065 880	1.58E+08
	worst	240	0	320 574.0	18 388 050	6 595 200	87 138 000
50_d0_tw4	best	3 780	158 880	229 692.0	61 954 032	154 742 400	89 737 200
	average	3 258	137 802	194 004.6	53 912 637	87 006 960	79 200 720
	worst	2 760	99 420	158 562.0	48 499 344	60 224 400	35 424 000
50_d1_tw1	best	3 000	17 160	443 322.0	1 810 542	58 820 400	0
	average	2 730	5 148	386 695.8	1 297 729	50 774 040	0
	worst	2 520	0	317 346.0	948 810	43 167 600	0
50_d1_tw2	best	1 740	0	235 548.0	2 880 042	86 929 200	0
	average	1 560	0	188 147.4	1 557 728	78 718 680	0
	worst	1 440	0	148 326.0	268 500	72 554 400	0
50_d1_tw3	best	1 080	0	366 858.0	25 042 092	18 925 200	2.55E+08
	average	768	0	324 304.2	24 065 212	16 047 360	2E+08
	worst	540	0	268 722.0	23 141 700	12 146 400	1.22E+08
50_d1_tw4	best	2 460	20 340	687 600.0	5 750 988	75 999 600	1.66E+08
	average	2 094	11 178	576 431.4	3 960 852	70 805 880	1.55E+08
	worst	1 680	0	496 482.0	2 033 028	61 135 200	94 795 200
50_d2_tw1	best	0	123 060	831 528.0	3 789 576	59 760 000	61 689 600
	average	0	122 784	694 749.6	3 531 262	49 678 920	33 022 080
	worst	0	120 300	594 426.0	3 149 166	36 532 800	0
50_d2_tw2	best	2 640	0	713 970.0	3 056 874	66 567 600	284 400
	average	2 484	0	630 510.0	2 398 305	59 207 760	145 440
	worst	2 220	0	551 298.0	1 336 314	44 132 400	0
50_d2_tw3	best	420	87 360	490 368.0	57 652 992	7 732 800	25 596 000
	average	342	87 096	458 721.0	49 416 319	5 318 280	15 331 680
	worst	240	86 040	376 998.0	51 309 015	2 350 800	0
50_d2_tw4	best	2 400	0	963 108.0	2 949 780	82 825 200	14 742 000
	average	2 226	0	900 396.0	1 458 745	50 275 080	5 660 280
	worst	2 100	0	821 442.0	600 840	29 865 600	0

Table 4.2: Hyperarea indicators of test problem results, 50 customers.

		Z1-Z3	Z1-Z5	Z2-Z3	Z2-Z5	Z4-Z3	Z4-Z5
250_d2_tw1	best	600	1 033 920	107 118.0	4.72E+08	11 491 200	2.7143E+10
	average	252	845 604	79 979.4	4.09E+08	8 209 800	1.8608E+10
	worst	60	599 640	53 874.0	3.55E+08	4 946 400	1.3184E+10
250_d2_tw2	best	540	1 029 300	227 088.0	4.9E+08	24 958 800	3.4807E+10
	average	276	820 344	173 223.0	4.44E+08	17 012 880	2.7858E+10
	worst	120	688 980	112 242.0	3.96E+08	10 735 200	2.3874E+10
250_d2_tw3	best	180	1 173 840	31 140.0	5.1E+08	3 661 200	2.2704E+10
	average	96	944 016	17 226.6	4.87E+08	1 946 880	1.7887E+10
	worst	0	775 740	0	4.54E+08	0	1.3895E+10
250_d2_tw4	best	420	1 485 540	89 472.0	4.08E+08	25 833 600	2.5134E+10
	average	173.3	1 281 972	61 505.33	3.82E+08	17 179 560	2.2787E+10
	worst	60	1 072 440	41 910.0	3.34E+08	9 165 600	2.0649E+10

Table 4.3: Hyperarea indicators of test problem results, 250 customers.

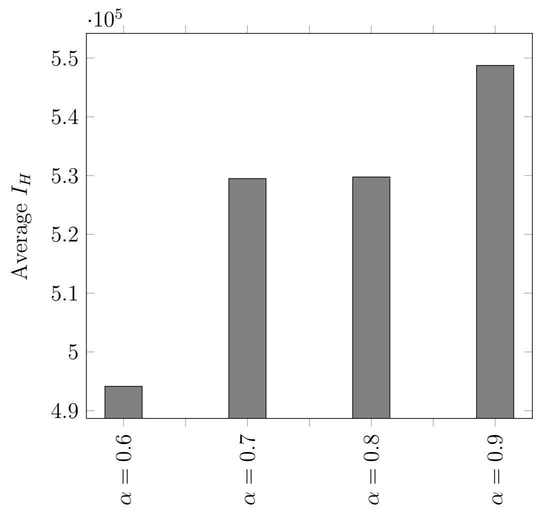

Figure 4.2: Variation of I_H for different α values.

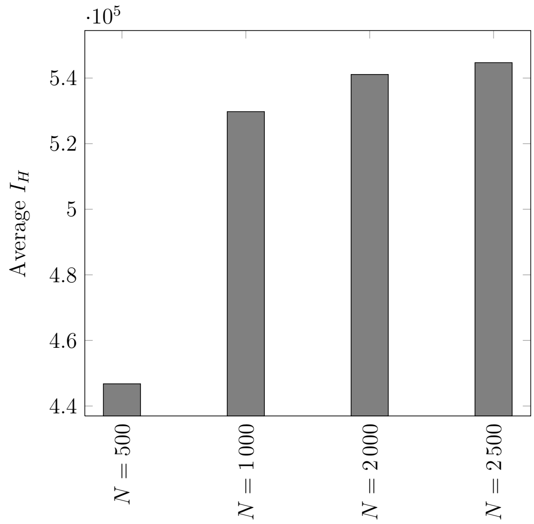

Figure 4.3: Variation of I_H for different N.

The results are shown in subsequent figures as follows: for each of the conflicting pairs of objectives, a figure showing the progression of the approximation front through the iterations (or loops) as explained in Algorithm 2 is presented. This serves as indicator of the worth of the CEM, as results improve from the completely random construction of routes in iteration 1 to the final approximation front formed in iteration 10. The adjacent figure shows the final approximation front in the objective domain as presented to the decision maker. Lastly, a table showing the routes for the set of vehicles ($\mathcal{V}_k$) of a particular solution point from the final approximation front, is presented. The tables (Table 4.4 to Table 4.9) list the reference numbers of the customers in the order that they were visited by each vehicle. These numbers follow from Castro-Gutierrez *et al.* (2011). In Table 4.4, for example, it is shown that six vehicles were used, and vehicle number 1 ($V1$) visited eight customers. Also, there are exactly 50 non-zero labels in the table, and the label sets represented by each column are mutually exclusive. The solutions listed here serves the purpose to aid in the discussion of the results (Section 4.4.5) and as a sample of the different benchmark problems. The rest of the solutions are documented in a similiar fashion in Appendix A.

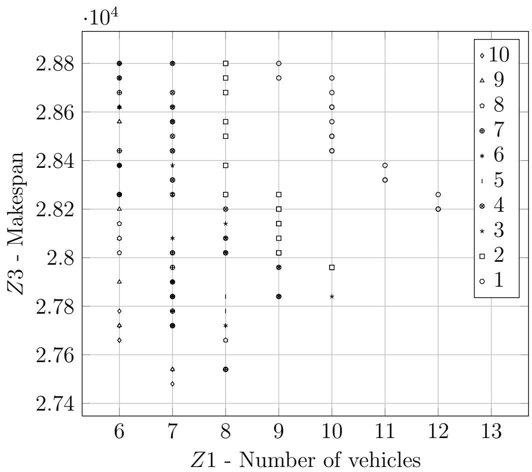

Figure 4.4: Front progression of 50_d1_tw4 for $Z1$ vs $Z3$.

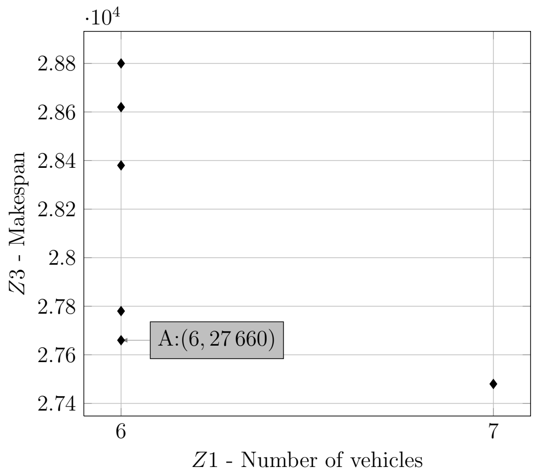

Figure 4.5: Final approximation front of 50_d1_tw4 for $Z1$ vs $Z3$.

4.4.5 Discussion of results

For all six cases the progression of the approximation front is evident and the worth of the CEM of estimating good solutions is illustrated. In the case of the discrete objective $Z1$, the number of vehicles, Figures 4.5 and 4.7 show approximate Pareto fronts that are limited in their multi-objective nature and exhibit seemingly redundant points at the lowest number of vehicles. As this is only evident in the case of the discrete objective, it can be deduced that the algorithm found a number of solutions for $Z3$ for the same value of $Z1$, and cannot justify fewer or more values for $Z1$, as no points between discrete values can be obtained.

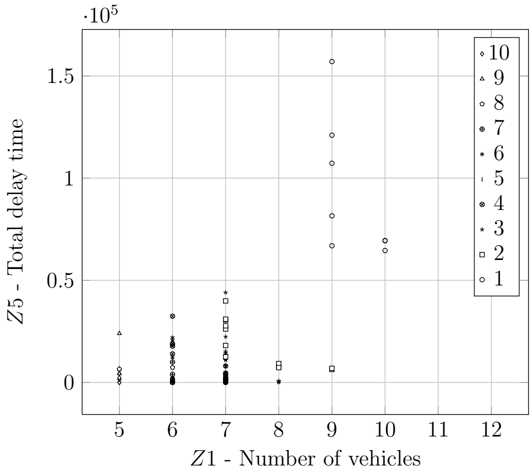

Figure 4.6: Front progression of 50_d1_tw4 for $Z1$ vs $Z5$.

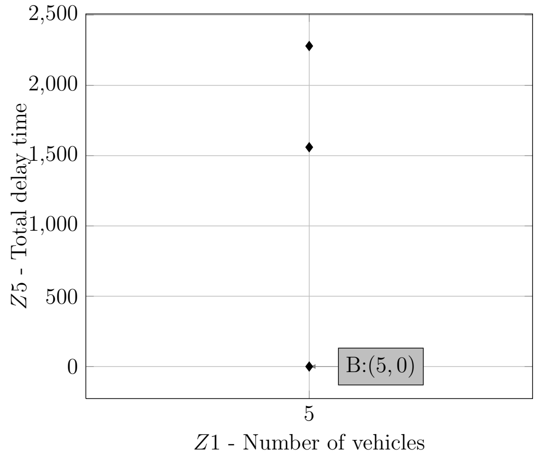

Figure 4.7: Final approximation front of 50_d1_tw4 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
649	1 725	2 000	1 173	2 104	1 870
1 384	856	430 148	430 030	1 714	2 107
430 761	1 888	1 897	1 703	430 378	1 777
2 044	1 362	1 813	1 781	948	1 203
2 121	1 678	1 463	669	430 625	1 389
2 149	2 073	486	2 003	0	1 875
430 804	1 588	1 509	0		974
482	1 686	661			1 235
0	2 152	907			2 138
	106	430 471			0
	0	1 721			
		2 007			
		0			

Table 4.4: Routes of solution A, 50_d1_tw4 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5
0	0	0	0	0
2 104	1 813	1 173	649	1 870
856	1 725	1 384	2 107	1 777
1 888	1 897	1 714	907	1 203
430 148	1 678	430 378	2 073	1 703
1 463	430 030	2 044	486	1 389
1588	974	1 875	1 509	669
106	430 761	1 235	1 686	482
661	1 781	2 121	2 152	0
430 471	2 149	2 138	1 362	
2 000	430 804	0	1 721	
430 625	2 003		0	
2 007	0			
948				
0				

Table 4.5: Routes of solution B, 50_d1_tw4 ($Z1$ vs $Z5$).

It is also valuable to note that in Figure 4.10 and, to a lesser extent, Figures 4.8, 4.12 and 4.14, the spread of the front becomes smaller as the algorithm progresses, and tends to converge to a limited region. This supports the finding in Section 4.4.3 that the number of loops (10) is adequate. If time permits, more outer loops can be added, but this would only lead to small improvements in the front. This choice is thus left to the decision maker, *i.e.*, to what extent priority is given to the execution time of the algorithm or to the precision of the solution.

Although Figure 4.15 exhibits a single-point solution (which is strictly not

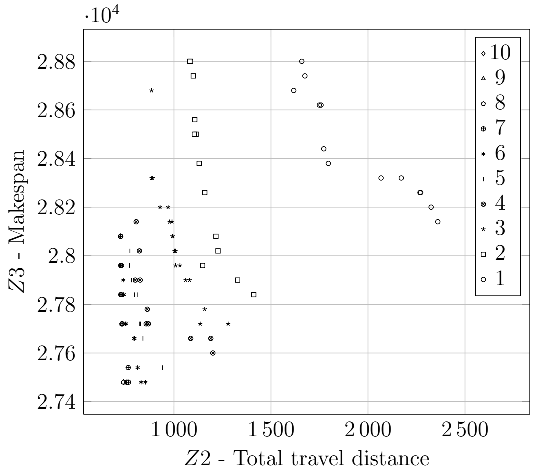

Figure 4.8: Front progression of 50_d1_tw4 for $Z2$ vs $Z3$.

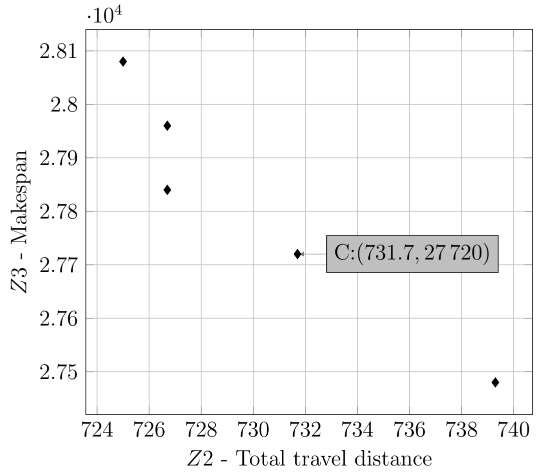

Figure 4.9: Final approximation front of 50_d1_tw4 for $Z2$ vs $Z3$.

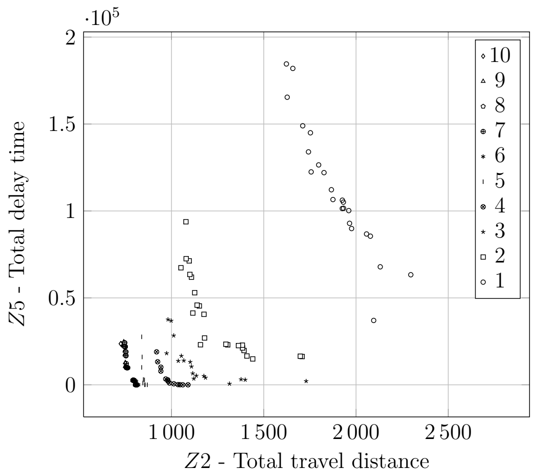

Figure 4.10: Front progression of 50_d1_tw4 for $Z2$ vs $Z5$.

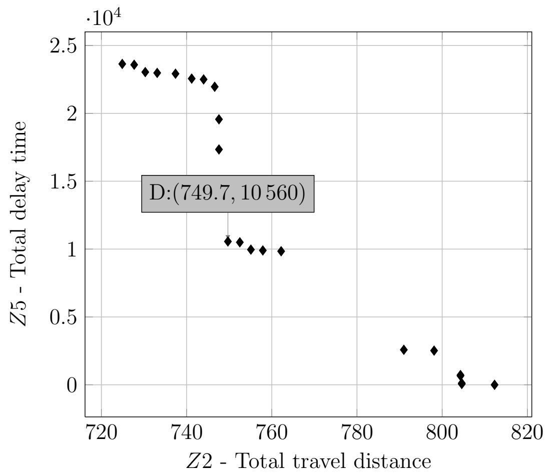

Figure 4.11: Final approximation front of 50_d1_tw4 for $Z2$ vs $Z5$.

Pareto anymore), Figure 4.14 illustrates that the non-optimal solutions are inherently multi-objective. The progression of the front to the absolute optimal (the best combination for $Z4$ and $Z5$ is 0) is good, and the multi-objective nature of the objective pair is not refuted.

The 250-customer problem poses challenges to the memory-allocation of typical desktop computers and the computational time, but one set of problems with the tightest capacity profile ("250_d2_tw1") was computed to investigate the feasibility of the CEM for multi-objective vehicle routing on larger scale problems. The

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
1 813	1 714	1 588	430 378	430 471	1 725
2 104	649	430 761	856	1 721	1 870
1 173	430 030	2 107	907	2 152	1 389
1 384	1 703	974	486	430 625	2 044
1 678	1 777	1 781	1 509	0	2 121
661	1 203	430 804	948		1 875
106	2 138	482	2 007		1 235
2 073	0	0	1 463		2 149
1 897			0		669
2 000					2 003
1 362					0
1 686					
430 148					
1 888					
0					

Table 4.6: Routes of solution C, 50_d1_tw4 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
1 725	1 870	1 714	2 104	856	1 588
2 107	2 044	430 378	1 813	1 173	1 463
430 761	2 121	106	1 384	649	2 073
1 781	1 389	661	1 897	430 030	430 625
669	1 235	486	1 888	1 703	2 000
482	1 875	907	430 148	1 777	0
0	974	1 509	1 686	1 203	
	2 149	1 678	2 152	2 138	
	430 804	2 007	1 721	0	
	2 003	948	1 362		
	0	430 471	0		
		0			

Table 4.7: Routes of solution D, 50_d1_tw4 ($Z2$ vs $Z5$).

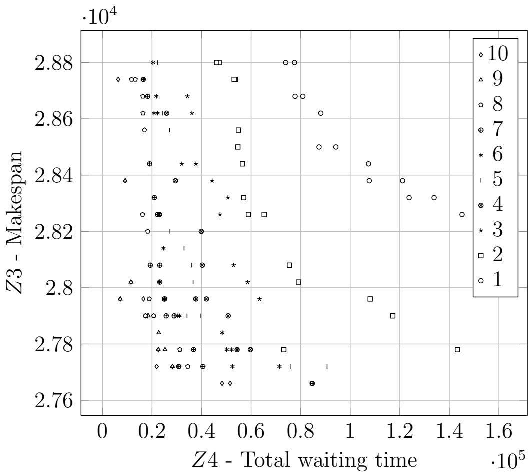

Figure 4.12: Front progression of 50_d1_tw4 for $Z4$ vs $Z3$.

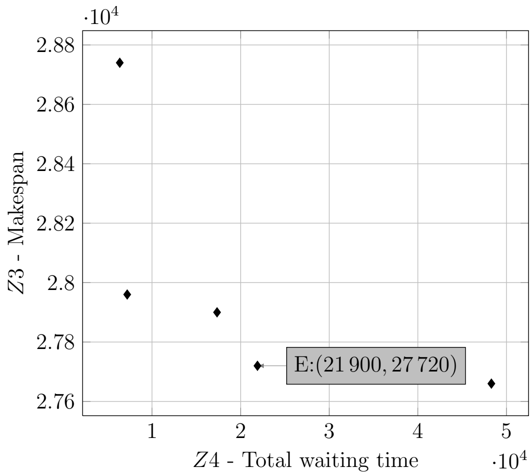

Figure 4.13: Final approximation front of 50_d1_tw4 for $Z4$ vs $Z3$.

method proved to be able to solve these problems and results showed similar characteristics to that of the 50-customer problems. The biggest difference can be seen in Figure 4.16 and Figure 4.17: in the 250_d2 problem objective $Z3$ exhibited the same behaviour as that of the discrete objective $Z1$, the number of vehicles. This is due to the nature of the problem and time windows, limiting the makespan of the longest route to a fixed number of possibilities. Figures 4.18 and 4.19 serve as an example of the $Z4$ and $Z5$ pairing, exhibiting a multi-objective approximation front as also found in the other 50-customer problems included in

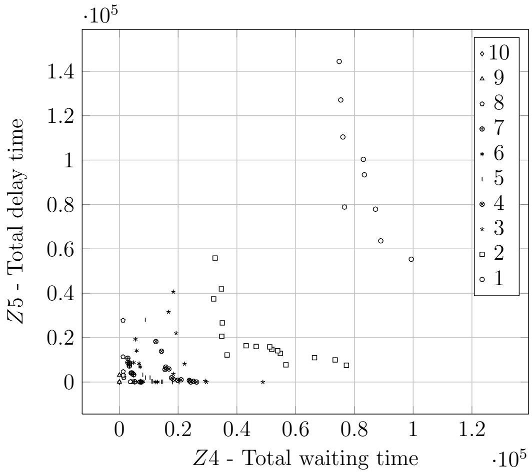

Figure 4.14: Front progression of 50_d1_tw4 for $Z4$ vs $Z5$.

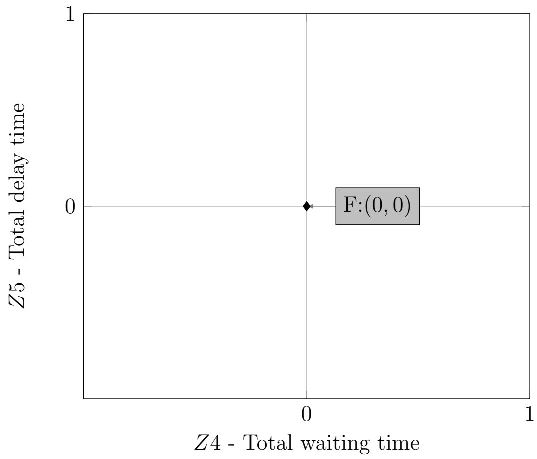

Figure 4.15: Final approximation front of 50_d1_tw4 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
2 107	2 104	1 725	1 777	649	1 870
1 897	430 378	1 888	2 000	1 714	430 030
1 703	2 121	974	2 044	1 678	1 389
1 235	907	1 588	1 686	1 362	1 509
2 149	856	1 781	2 152	1 384	1 721
430 804	669	2 138	106	486	430 625
482	2 003	0	0	2 073	430 471
0	0			948	0
				661	
				2 007	
				1 463	
				0	

Table 4.8: Routes of solution E, 50_d1_tw4 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
649	2 107	1 725	1 870	1 897	1 173
430378	1 777	1 384	2 000	1 813	856
1203	106	2 104	1 875	1 714	1 888
661	1 463	1 703	2 044	430 030	430 761
2149	1 509	1 588	2 121	974	1 235
430471	907	669	1 686	1 389	2 073
430625	486	2 003	2 152	1 721	430 148
0	1 781	0	1 362	2 007	2 138
	430 804		0	948	0
	482			1 678	
	0			0	

Table 4.9: Routes of solution F, 50_d1_tw4 ($Z4$ vs $Z5$).

Appendix A.

The discussion of the results is concluded by presenting the nature of the routes in a solution on a coordinate map with the depot at $(28.0718, -16.622)$. Solution A (indicated in Figure 4.5) is presented in Figure 4.20 and Solution F (indicated in Figure 4.15) in Figure 4.21. The complex nature of the multi-objective VRPSTW is illustrated since a comparison of different objectives yields completely different routing solutions. Figures 4.22 – 4.25 are an illustration of the routes of one solution of the 250_d2_tw1 problem for objectives $Z4$ vs $Z5$, shown as Solution G in Figure 4.19. The solution consisted of 37 routes which are displayed over four

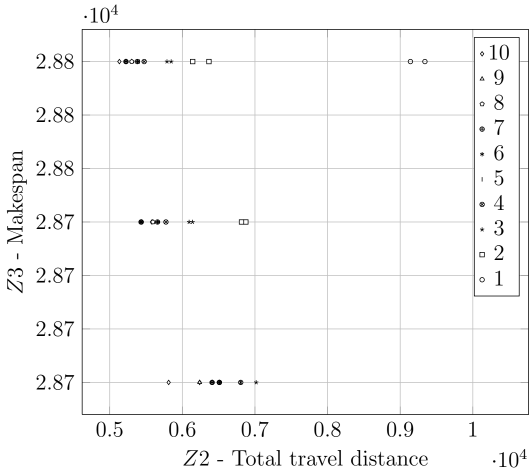

Figure 4.16: Front progression of 250_d2_tw1 for $Z2$ vs $Z3$.

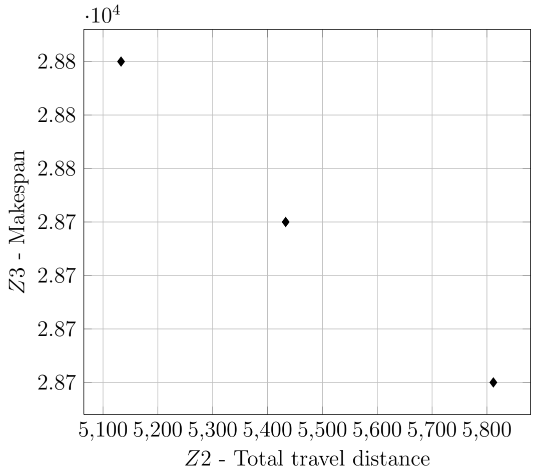

Figure 4.17: Final approximation front of 250_d2_tw1 for $Z2$ vs $Z3$.

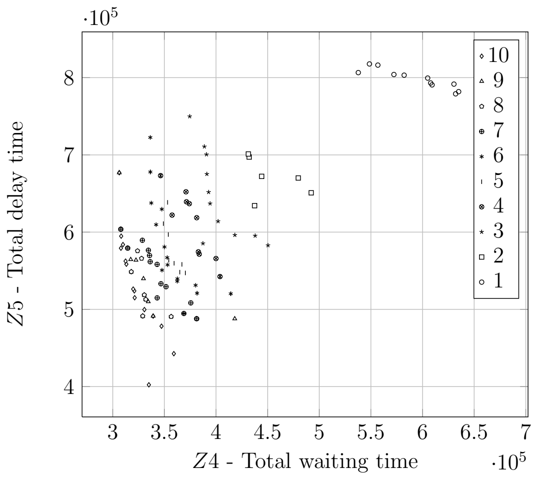

Figure 4.18: Front progression of 250_d2_tw1 for $Z4$ vs $Z5$.

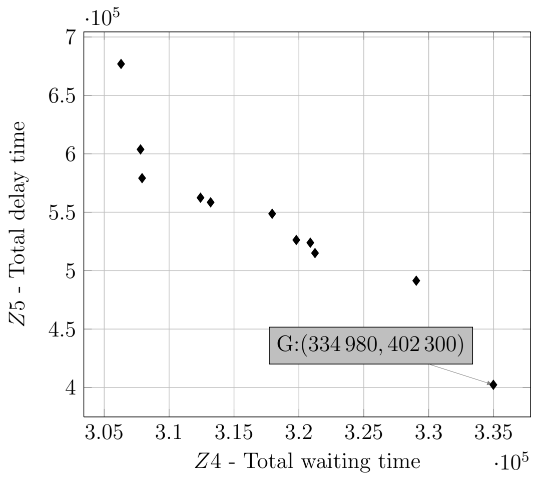

Figure 4.19: Final approximation front of 250_d2_tw1 for $Z4$ vs $Z5$.

figures of nine or ten routes each to provide clarity.

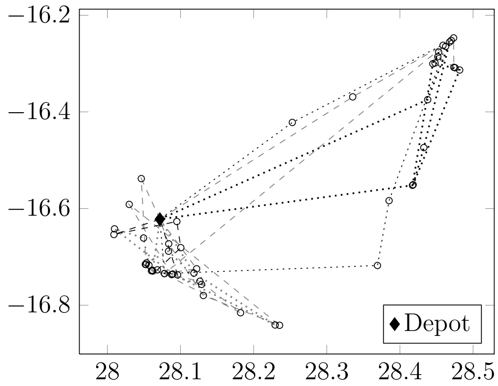

Figure 4.20: Map of routes of solution A, 50_d1_tw4 for $Z1$ vs $Z3$.

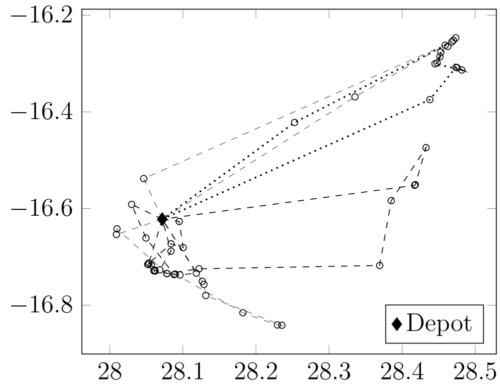

Figure 4.21: Map of routes of solution F, 50_d1_tw4 for $Z4$ vs $Z5$.

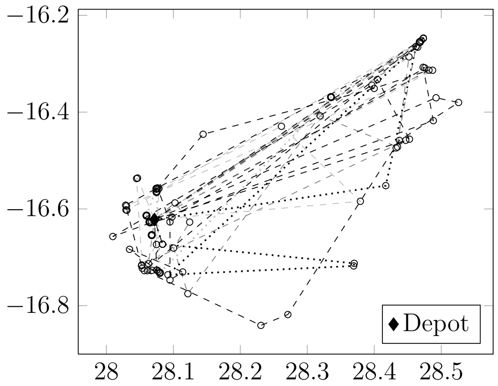

Figure 4.22: Map of routes of solution G, 250_d2_tw1 for $Z4$ vs $Z5$, Part 1.

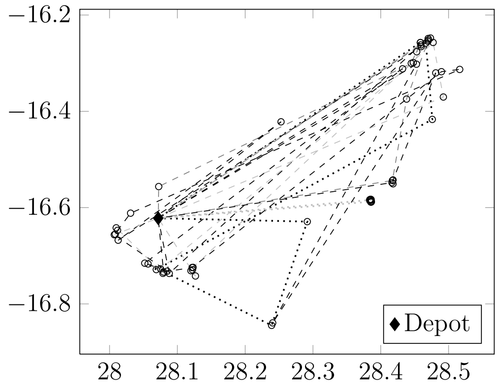

Figure 4.23: Map of routes of solution G, 250_d2_tw1 for $Z4$ vs $Z5$, Part 2.

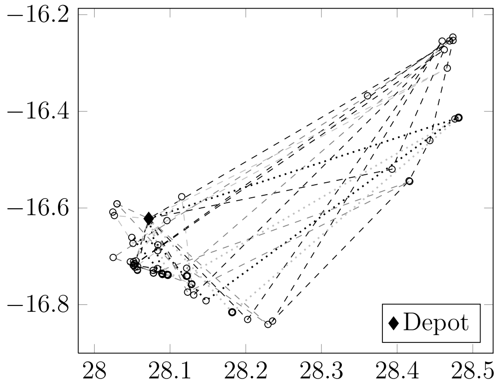

Figure 4.24: Map of routes of solution G, 250_d2_tw1 for $Z4$ vs $Z5$, Part 3.

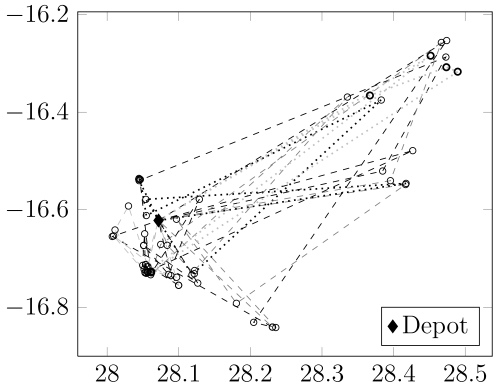

Figure 4.25: Map of routes of solution G, 250_d2_tw1 for $Z4$ vs $Z5$, Part 4.

4.5 Concluding remarks on Chapter 4

The aim of the chapter was to apply the CEM for MOO to the VRPSTW and subsequently provide an alternative approach to the multi-objective VRPSTW. It was found that the CEM for MOO adequately obtains an approximation set that progresses towards the Pareto front of solutions for conflicting objectives of a given problem. In addition a set of solutions for the new benchmark problems introduced by Castro-Gutierrez *et al.* (2011) as found by the CEM for MOO is documented for the first time to contribute to the literature available in the field of the multi-objective VRPSTW.

The speed performance of the CEM itself proved good but the route-construction algorithm (as explained in Section 4.3) is a definite bottleneck. Future work should look at streamlining the construction of routes with a transition probability matrix as required by the CEM. In addition, a population size of $N = 2\,500$ and 250 customers required too much memory to be executed on a typical personal computer (3 Gb RAM), therefore further research is needed to find a less memory-intensive representation of the decision variables (routes).

The chapter provided support for the use of the CEM for MOO in the largely theoretical VRP, further securing the merit of the proposed method. The next chapter investigates the use of the method in the case of a real world problem that is modelled with simulation, the blood inventory and supply chain problem.

CHAPTER 5

BLOOD SUPPLY CHAIN AND INVENTORY MANAGEMENT

The research aim is based on the need for evidence of the worth of the cross-entropy method (CEM) for multi-objective optimisation (MOO) in the field of combinatorial optimisation. The previous chapter investigated the use of the method in the largely theoretical field of vehicle routing. This chapter is dedicated to the application of the method to a real world case study in the context of blood supply chain management.

The management of blood is of great importance to the medical discipline and a problem that has seen an increase in studies over the past years. Blood is an unique commodity due to a number of factors: it is classified as a perishable product, the supply originating from donors is stochastic, the demand is also stochastic, as apart from scheduled operations it is dependent on the occurrences of trauma, and finally the products are classified into the relevant blood groups and Rhesus factors. This chapter starts with an overview of the field of blood management and the scholarly literature pertaining to the case study. The case study at the Western Province Blood Transfusion Service (WPBTS) is then introduced and discussed, and finally the optimisation of the system is documented.

5.1 Research design

The research documented in this chapter has the purpose to model a real world system using discrete event simulation and to solve a multi-objective problem(MOP) instance with the CEM. The attainment of this research aim is driven by the following objectives.

- Explore the literature available on blood supply chain and inventory management to get a grasp of common problems, system designs and methodology.
- Do a case study of the WPBTS through interviews, observations and other research and build a conceptual model of operations.
- Master the use of the Simio® simulation software and build and validate a simulation model of the WPBTS problem.
- Modify the cross-entropy method for multi-objective optimisation of the discrete WPBTS inventory problem in Matlab®.
- Integrate the simulation model and optimisation algorithm using the Simio Application Programming Interface (API) to access the model back-end in Matlab® and validate the combined optimisation model.
- Conduct experiments to determine optimal inventory policies in the case of a perfect supply source, in the case of a limited supply and to determine the effect of a transportation costing structure.
- Document and interpret results.

5.2 Literature introduction

The literature of blood management is varied in applications, problem definition and scope. Beliën & Forcé (2011) provide a literature review of both inventory and supply chain management of blood products. The paper shows that interest in the field peaked during the period of 1976 – 1985 with a substantial number of papers being published, but this decreased dramatically before a renewed interest occurred in the past decade. In South Africa the studies on the management of blood are limited; Le Roux (1981) used mathematical techniques to determine the optimal location of blood banks.

The literature available is quite extensive, but provides an interesting background to the study conducted. The aim of this section is to introduce certain concepts and previous work in the field to aid in the documenting of the reasoning and development of the case study. To this end the section is subdivided in focus areas that will assist in modelling the WPBTS system.

5.2.1 Blood products and groups

Blood is a rather complex substance, consisting of a combination of fluid and different kinds of cells. Blood can be characterised in four different groups, namely A, B, AB and O. In addition to being classified in these groups, blood is also characterised by a Rhesus factor (RhD), a protein that is present (or absent) on the surface of red blood cells (Chapman *et al.*, 2009). The combination of these classifications results in the following blood groups: A^+, A^-, B^+, B^-, AB^+, AB^-, O^+, O^-. These eight blood groups determine donor-recipient compatibility. Table 5.1 shows the distribution of blood groups among blood donors in South Africa according to the SANBS website (`www.sanbs.org.za`, Accessed: 30 July 2012). O^- blood is compatible with all other blood groups and can be used in an emergency. In turn people with AB^+ blood type can receive blood from all blood groups (Alexander, 2012). Donated blood is either transfused as whole blood (WB) or divided into the four constituent parts: packed red blood cells, white blood cells, plasma and platelets. Red blood cells are used to transport oxygen and carbon dioxide in the body, white blood cells work against viruses and infections, plasma is the liquid component of blood containing nutrients and protein and finally platelets help the blood clot (`www.wpblood.org.za`, Accessed: 30 July 2012). According to Allen (2001), red blood cells are obtained by sedimation or centrifugation of whole blood and the removal of the plasma portion. Red blood

	O	**A**	**B**	**AB**
Positive	39%	32%	12%	3%
Negative	6%	5%	2%	1%

Table 5.1: Distribution of blood groups among blood donors in South Africa.

cells are also frequently modified to address specific transfusion requirements, eg. by washing, irradiation and leukoreduction. The latter is "the removal of white blood cells to prevent allo-immunization and viral disease transmission and to reduce the occurrence of febrile transfusion reactions" (Allen, 2001). A large number of products form part of modern transfusion science, including platelets (with a short shelf life), granulocytes, fresh frozen plasma and cryoprecipitate to name a few. The three products under consideration in the case study of the WPBTS with the corresponding shelf lives (Allen, 2001), are the whole blood (1 year frozen), concentrated red blood cells (42 days) and finally the leuko reduced red blood cells (42 days).

5.2.2 Perspectives and characteristics of the field

This subsection discusses the different perspectives from which blood management is studied and mention some of the characteristics that makes it such a complex system. Although the commodity of blood is primarily handled by physicians, other healthcare professionals and scientists, the processes concerned with the distribution and other logistics introduce literature to the field that is varied in its origin. The increase in the cost of healthcare and in the demand for good service in modern times have led to the development of disciplines such as health systems engineering, in which the health industry is considered as a complex system in which engineering applications are continually being identified and applied. Katsaliaki (2008) supports the use of these applications to the field of blood management, "Better stock management in hospital blood banks can be achieved with the implementation of rules that usually apply to manufacturing and other profit-driven companies, such as updated optimal stock levels, determined ordering points, specified ordering, delivering policies *etc.*" In addition, Pierskalla (2005) notes that the blood supply chain is unique as it can and should be examined as a whole system and not as subsystem of a larger system, as occurs in other supply chains.

With the study conducted from the perspective of health systems engineering, the literature reviewed in this section is scoped accordingly. The system of the blood supply chain is complex and the use of both health engineering concepts and systems analysis are supported. As early as 1979, Prastacos & Brodheim (1979) states that efficient management of the regional blood resources is a complex task. A list of reasons include, as mentioned before, the perishable nature of the product, the probabilistic and seasonal supply and demand, and finally the special conditions in which components are inventoried. Most countries use a voluntary blood donation system, and despite some dedicated and regular donors, the influx of donations (or the supply) is subject to a number of factors and consequently stochastic. Chapman *et al.* (2009) state that in the UK the use of blood during surgery has decreased significantly due to several factors, such as improved techniques and other advances in the medical field, but this demand for blood products still exhibits considerable variability.

The blood inventory problem is complex with certain characteristics inhibiting the easy optimising of the system. Kendall & Lee (1980) emphasise that the regional blood inventory problem is complex due to multiple objectives, multiple inventories and of course the perishability of the product. Jennings (1973) states that the characteristics of whole blood inventories defeat analytical approaches to such a study, making reference to the modelling of inventory as a 21 dimensional variable due to the possible age of the specific blood product. This supports the need of simulation to approach the problem. Simulation as a health systems problem solving methodology is further discussed in Section 5.4.1.

5.2.3 Optimisation and performance measures

Blood supply chain management is similiar to so many other disciplines as stakeholders are always in search of improvement. In the case of optimisation, various objectives or performance measures have been identified and used in literature. Some objectives or performance measures include outdate rates and shortage rates (Prastacos & Brodheim, 1979), reduced costs and increased safety, fewer group substitutions, deliveries and mismatching (use of another compatible blood group) (Katsaliaki, 2008), minimising the quantity of blood imported from outside the system and the range of quantities of blood assigned daily from within the system (De Angelis *et al.*, 2001). Gregor *et al.* (1982) further look at the number of emergency orders placed at the regional centre, the number of routine surgeries postponed and delivery costs. Cooper (2001) cites a key performance measure of any perishable inventory system is the number of items that must be discarded because it became unusable when its lifetime ended. Yegul (2007) defines performance measures as the inventory level, the outdates, cross-matching and shortages rates and also the number of scheduled and *ad hoc* deliveries. When assessing the entire supply chain of blood, another metric derived from classic supply chain management is the backorder percentage or cost, *i.e.*, the amount of orders that can not be supplied by the current inventory level at blood banks (Rytilä & Spens, 2006). In the study presented in this chapter, another classic inventory management metric – the service level – is used along with the distance travelled or the travel cost accrued to enable the delivery of blood products. The arguments supporting this decision is documented in Section 5.5.2.

With such a large number of metrics, multi-objective optimisation becomes a serious consideration. The management of this unique commodity is often concerned with conflicting objectives and needs, as can be seen in the few examples listed in the previous paragraph. Chapman *et al.* (2009) feels that this is a particular issue in the case of inventory management, as blood services are faced with the challenge to balance the need for sufficient stock in order to meet demand with the issue of wastage due to excess stock levels.

MOO in the field of blood supply chain management is conducted by De Angelis *et al.* (2001) in formulating a MOO linear programming problem and solving it through a lexicographic minimisation process. They document the scheduling of operations within tolerance limits in such a way that the amount of blood orders that are scheduled from outside the system is minimised. In addition blood is distributed more evenly during the month.

Jennings (1973) investigates blood bank inventory control with specific reference to the interaction of hospital blood banks on a regional level. Although not defining it as such, the problem becomes multi-objective with conflicting objectives in the operating curve; the percentage of outdated products versus the shortage percentage. These values change for different values of inventory levels at the particular hospitals. Another min-max operating curve is shown in the comparison of objectives to minimise the inter-hospital shipments per bank and to maximise the percent reduction in shortage and outdating that change as the number of participating hospitals change. In addition these curves are compared for different policies (Jennings, 1973).

5.2.4 Stakeholders and environments of the blood management system

Because the ownership of an entity of blood changes hands at various places in the supply chain, literature is generally divided into the viewpoint from which the supply chain is studied. Regional centres, donor centres, blood banks and hospitals are all role players in the management of the blood. In order to ensure good community medical care the supply of the valuable commodity of blood needs to be managed, and Gregor *et al.* (1982) investigate the workings of an organised regional blood distribution system. A regional blood-distribution policy has to consist of two important components as identified by Gregor *et al.* (1982): "a set of optimal inventory levels for each regional hospital and a procedure for scheduling deliveries to the hospitals." These components put forward by the literature is paramount to the rest of the study. In the case study of the WPBTS, an optimal regional blood-distribution policy is investigated by setting the inventory levels and the reorder points at the regional hospitals (in effect determining delivery schedules).

At a hospital level, good management of the blood system can be employed on a day to day basis by planning medical and surgical interventions according to urgency and availability, where necessary placing the blood requests that are not so urgent on hold *etc.* (De Angelis *et al.*, 2001). A systems view is also encouraged with Katsaliaki (2008) suggesting the use of an online, real-time stock control and ordering system for all the blood banks which will enable coordinated supply and demand. The idea is that information on inventory levels and ordering processes will be transparent to all members of the system. These viewpoints also affect the objectives of the study. Depending on the vantage point of the decision maker, objectives will carry different weights. The case study presented in this chapter is conducted from the viewpoint of the central distribution centre, encouraging a systems view approach.

The supply chain also stretches over various environments which have an impact on the inventory level of the blood. These include factors that influence a hospital's red cell inventory levels such as size, the distance from the blood bank, and the presence of specialist clinical units such as trauma and orthopaedics

(Chapman *et al.*, 2009). Katsaliaki (2008) explains that the product incurs losses due to inventory management, transfusion practices and outdating.

In addition to inventory management and assignment policies the supply chain of blood products is concerned with delivery strategies to ensure the timely delivery for emergency and scheduled stock. Hemmelmayr *et al.* (2009) investigates alternative strategies for the cost-effective delivery of an Austrian blood bank. The combination of integer programming and the variable neighborhood search approach is used to evaluate the difference in cost of the use of fixed routes or a more flexible routing solution based on regularity of delivery.

5.2.5 Conclusion of the literature introduction

With these four broad categories in mind, a background on the case study is provided. The study is conducted with a healthcare engineering approach and many of the characteristics of a complex system are present in the studied system. The aim of the study is to optimise two objectives simultaneously, and the focus is on the central distribution centre as stakeholder and the environment restricted to the regional blood bank level. These aspects are discussed further in the next section on the WPBTS, and more detail will be provided as the study is documented. In conclusion, Rautonen (2007) summarises the three components of an efficient supply chain for blood supply as follows: Cost efficient and reliable, blood delivery managed according to hospital needs, and optimised internal management of the blood supply system. With these components and the overview of literature in mind, the next section documents the case study of the WPBTS and the conceptual model of the blood supply chain in the Western Province before the chapter continues with the aim to optimise these components.

5.3 The Western Province Blood Transfusion Service

In South Africa blood transfusion is administered by two bodies, the South African National Blood Service (SANBS) and the Western Province Blood Transfusion Service (WPBTS). The SANBS provides service to most of the regions while the WPBTS is only responsible for the greater Western Province region. Due to geographical location and the scoping of the project the study was conducted with the WPBTS as subject. The WPBTS is a non-profit, independent organisation that supplies safe blood and blood products to the greater region of Cape Town reaching to Worcester, Paarl and George. The WPBTS is dependent on donations of the provincial community, ensuing a supply that is often volatile. Donations are bled throughout the province and transported to a central facility in Pinelands, Cape Town where the necessary testing and processing of the products are done. Depending on the processes required, the products are then made available to a central stock of supply from where it is shipped to seven regional blood banks. These blood banks are responsible for distributing blood as required to their dependent hospitals.

Allocation of the blood from the central supply is decided on a daily basis by the staff of the WPBTS by reviewing the required inventory level and the current stock at the blood banks (Alexander, 2012). In general, blood is shipped on a daily basis to blood banks with a high turnover and to some of the smaller banks every other day. This is however not a general rule, as decisions are made as necessary. A second shipping on a day is also possible if there is a big deficit at a blood bank, or if the donations are only made available later in the day.

Travelling large distances is often necessary due to the dispersed nature of the blood banks, and rising cost in fuel has led the student to hypothesise that a simplified distribution system can be advantageous. In addition a formalised inventory policy is put forward that minimises the effect of the expertise of the decision maker. An inventory system is defined by Jacobs (2004) as the set of policies and controls that monitor levels of inventory and determine what levels should be maintained, when stock should be replenished and the size of the orders. This study has the aim to develop such an inventory system or policy with regard to the inventory levels and reorder points at the regional blood banks. In order to aid the study and scope it to the time available a few simplifications were made, but the complex system lends itself to further research.

- Data available was restricted to the daily stock reports from the regional blood banks. The actual demand as experienced at the separate hospitals was deemed beyond the scope of the study. The fluctuation in inventory levels at the blood banks serves as a summation of the demand at the hospitals (final customer).

- No data was made available on the number of donations and products produced from the WPBTS. This is unfortunate as the volatile supply is a definitive characteristic of the blood supply problem. Despite this obstacle, the study investigates the influence of a capped supply on the performance measures in Section 5.6.

- Blood products investigated include whole blood (WB), leuko depleted red cells (LDRC) and concentrated red cells (RCC) due to data available. Blood platelets have a much shorter lifetime and are in general ordered as needed from the central supply. This also leaves room for an interesting further study.

- Delivery to the blood banks is done by a third party courier who travels the route to deliver other products as well. The study is conducted from the viewpoint that delivery vehicles only carry blood products that form part of the inventory policy. It is assumed that a suggestion on an improved delivery policy can initiate a review on the schedule of the other delivered products.

The blood banks that form part of the WPBTS and the proposed inventory policy and the acronyms used in the rest of the document are listed in Table 5.2.

Symbol	Description
DC	Central Distribution Centre (Pinelands)
GEO	George
GSH	Groote Schuur Hospital
MCV	Vergelegen Mediclinic (Somerset-West)
PAARL	Paarl
RCX	Red Cross Children's Hospital
TBH	Tygerberg Hospital
WTR	Worcester

Table 5.2: Acronyms of blood banks.

The blood products for which an inventory policy is investigated are limited to three types, namely the whole blood product (WB), the red cell concentrate (RCC) and finally leukocyte depleted red cells (LDRC). With a total of seven blood banks, three products and eight blood groups, this cumulates to $7 \times 8 \times 3 = 168$ different inventory values. Due to the current policy of not holding inventory of certain combinations of products and blood types and the fact that no data is subsequently available on the demand from the blood banks, the number of inventory values investigated were limited to 97. In combination with the reorder points this results in a problem with 194 decision variables. Future work can consider incorporating the products for which data was not available.

5.4 Simulation as modelling methodology

As explained in the first section of the chapter, the management of a blood inventory and distribution system can be complex and difficult to model. The problem presented in the previous section poses various challenges, most notable the stochasticity of supply and demand of blood in the Western Province and the large number of system components. This section documents the methodology followed to model this system by providing insight from literature and discussing the applicability of the method to the system. Simulation is introduced as a method used in health systems engineering, the reason for settling on this method is documented and the application of the general method to the specific problem is then explained in the next section.

5.4.1 Simulation in healthcare

Simulation is one of the techniques in the field of Industrial Engineering that contributes to the improvement of systems. Kelton *et al.* (2010) provide the following definition: "Computer simulation is the imitation of the operation of a system and its internal processes, over time and in appropriate detail to draw conclusions about the system's behaviour." In the field of healthcare engineering the use of simulation to model processes is widely documented. In addition to the high cost of healthcare, simulation modelling is favoured as experiments on a real-life, real-time system is not feasible when working with human lives. Healthcare systems are also generally stochastic and complex, encompassing a wide variety of role players, resources and time-dependent events. In their literature review of blood product supply chain management, Beliën & Forcé (2011) group papers according to the solution method used and it is found that simulation and statistical analysis are used more often than other methods such as linear programming, queuing models and dynamic programming.

With regard to the applicability of the method to the field of blood supply chains, Beliën & Forcé (2011) state that the regular use of simulation confirms the complexity of the problem and it has the advantage that it enables detailed results. Disadvantages of the method according to Beliën & Forcé (2011) is that reported results are dependent on a specific situation and can not be generalised, and more importantly the method does not provide any guarantee of optimality. To a certain extent, the use of an optimisation model (the cross-entropy method) in conjunction with the simulation model as proposed in this study eliminates this disadvantage.

In the field of simulation of the blood supply chain, Gregor *et al.* (1982) did four experiments, altering the initial inventory of the system, changing the number of delivery vehicles, altering the role of the regional centre by changing how much control of the inventory it retains and finally altering the age constraints on hospital's inventories. Katsaliaki & Brailsford (2006) simulate the blood supply chain in the UK to compare different management policies such as the time of placing orders, the *ad hoc* ordering policy, the number of days of stock kept, *etc*. The comprehensive study then examines the effect of the policies on a number of objectives. Rytilä & Spens (2006) also use discrete event simulation to compare a number of scenarios in the supply chain of Finland. These and other studies show that the use of simulation is greatly supported in healthcare management and blood management and contributed to the initial ideas on this case study.

5.4.2 Simulation as a problem solving technique to the WPBTS

Although simulation has gained immense support in various fields such as healthcare due to its numerous advantages, care must be taken before selection as a problem solving technique due to the time consuming nature of such a study. Disadvantages documented by Robinson (2004) include the fact that building a model is time consuming, the technique is data hungry and there exists the possibility of a lack of validity. All of these and other disadvantages such as the cost of a simulation package suggest that the use of simulation as a solution technique should be well supported by the nature of the problem, in fact, Pidd

(1998) recommends that simulation be used as a last resort in modelling. That being said, the following paragraphs provide support for the use of simulation in the case of the WPBTS by matching literature on the need for simulation to the nature of the problem.

Robinson (2004) answers the question "Why Simulate" from the perspective of three aspects of operations systems: its variability, interconnectedness and complexity. *Variability* is present in many aspects of daily life, even more so in the case of a blood supply chain. The need for blood is difficult to forecast and dependent on varying conditions or events. In addition the donations of blood exhibit variability over a period of time.

In the case of combinatorial optimisation, operations systems under study introduce a *combinatorial complexity.* This particular complexity is related to the number of components in a system or the number of possible combinations of system components (Robinson, 2004). The blood transfusion problem in general and the particular case of the WPBTS are subject to various system components. The type of blood products that is in demand is varied in the type of product as well as the number of blood groups. If the Rhesus factor is considered along with the blood groups, blood products are supplied as eight different entities. For the purpose of this study, the different blood products in demand at the hospitals are restricted to three. The WPBTS supplies blood to seven regional blood banks that each serve a number of hospitals, subject to different variations. The combination of blood products and blood banks as components of the greater operations system result in a combinatorial complexity that calls for the use of simulation.

Finally, operations systems that require simulation are characterised by being *interconnected.* Components in the system affect one another and a change in variables influences other parts of the system. The WPBTS has a central blood supply from which the different blood banks and hospitals are served. Due to the limited nature of the supply of blood, a blood distribution policy is found in the interconnectedness of the different inventories at blood banks. An increase in demand at one hospital will immediately lead to a depleted stock level at the central blood supply that affects all of the blood banks. It is clear that the blood supply system in the Western Province is characterised by all three factors as defined by Robinson (2004), which imposes the need to use simulation.

5.5 Formulation of the WPBTS model

The previous sections provided an overview of the literature in the field of blood supply chain management and the use of simulation in modelling a real world system to aid optimisation. This section discusses the building of such a simulation model in the context of the WPBTS for the purpose of this study and the application of the CEM for optimisation (Chapter 3).

5.5.1 Problem model of the WPBTS case study

White & Ingalls (2009) explain that a model is an abstraction of reality but generally simplified to only include detail and a scope that is necessary to study certain variables and/or objectives. The blood supply chain is a complex system with many components and details, but due to time limitations, the abstraction of reality that is investigated is concerned with the inventory management at the regional blood banks. In addition the model is subject to a few assumptions that was introduced in Section 5.3. Other assumptions will be explained as the building of the model is described in this section. The simulation model is built to examine the inventory policy of the WPBTS and to determine how this affects both the perceived service level at the blood banks and the demands it places on the delivery system. With this objective in mind, the model is scoped accordingly.

The simplified version of the modelling process as found in Sargent (2005) serves as a starting point for the discussion of the methodology followed in modelling the blood product supply chain. The process is founded on three entities, the problem entity (Section 5.3), the conceptual model and the computerised model.

5.5.1.1 Conceptual model

Analysis and modelling to obtain a conceptual model from the real world system was done by a number of visits and emails to the WPBTS head offices in Pinelands, and a visit to the Mediclinic Vergelegen Clinic blood bank in Somerset-West. The assumptions made were explained in Section 5.3.

A conceptual model is necessary to establish the model logic that will enable the study to be conducted. In addition, a conceptual model identifies the necessary input data and the typical output that should be investigated. The inventory system at the central distribution centre of the WPBTS can be described as a combination of a Q-model (fixed-order quantity system) and a P-model (fixed-time period reordering system) (Jacobs, 2004). Inventory is reviewed daily and although there is no fixed reorder point, staff determine whether to replenish stock at the different blood banks, effectively establishing a reorder point (Alexander, 2012). The inventory replenishment scheme in the conceptual model as used in Simio® is derived from Jacobs (2004). The process flow diagram in Figure 5.1 is an illustration of the basic conceptual model. The model is subdivided into three role players, the regional blood banks, the central distribution bank and the distribution vehicles. One of the important aspects of building a conceptual model is to establish the input data required and the output data to record. These requirements are shown in Figure 5.1 and Section 5.5.1.3 describes the data acquisition process that followed the conceptual model.

Sargent (2005) defines conceptual model validation as determining whether the structure and logic of the model is a reasonable representation for the purpose of the study. As the conceptual model was built after interviewing stakeholders to the system on a number of occasions, the validation and creation of the model was done simultaneously. The conceptual model as understood by the student was confirmed during the interview process and observations.

5.5.1.2 Computerised model

> *Fully mastering the art of building intelligent objects involves learning the events and the collection of available process steps, along with the knowledge and experience of how to combine these steps to represent complex logic* (Pegden, 2012).

The computerised model was built in the Simio® software environment, employing an object-oriented programming method. According to Pegden (2012) there are six basic classes of objects in Simio®. Four of these classes are used to model the WPBTS system, *fixed objects* (in the case of the central distribution centre and the regional blood banks), *entities* (orders that represent the combination and number of blood products), *links and nodes* (to build the network of delivery between the blood banks) and finally a *transporter* (the delivery vehicle). The

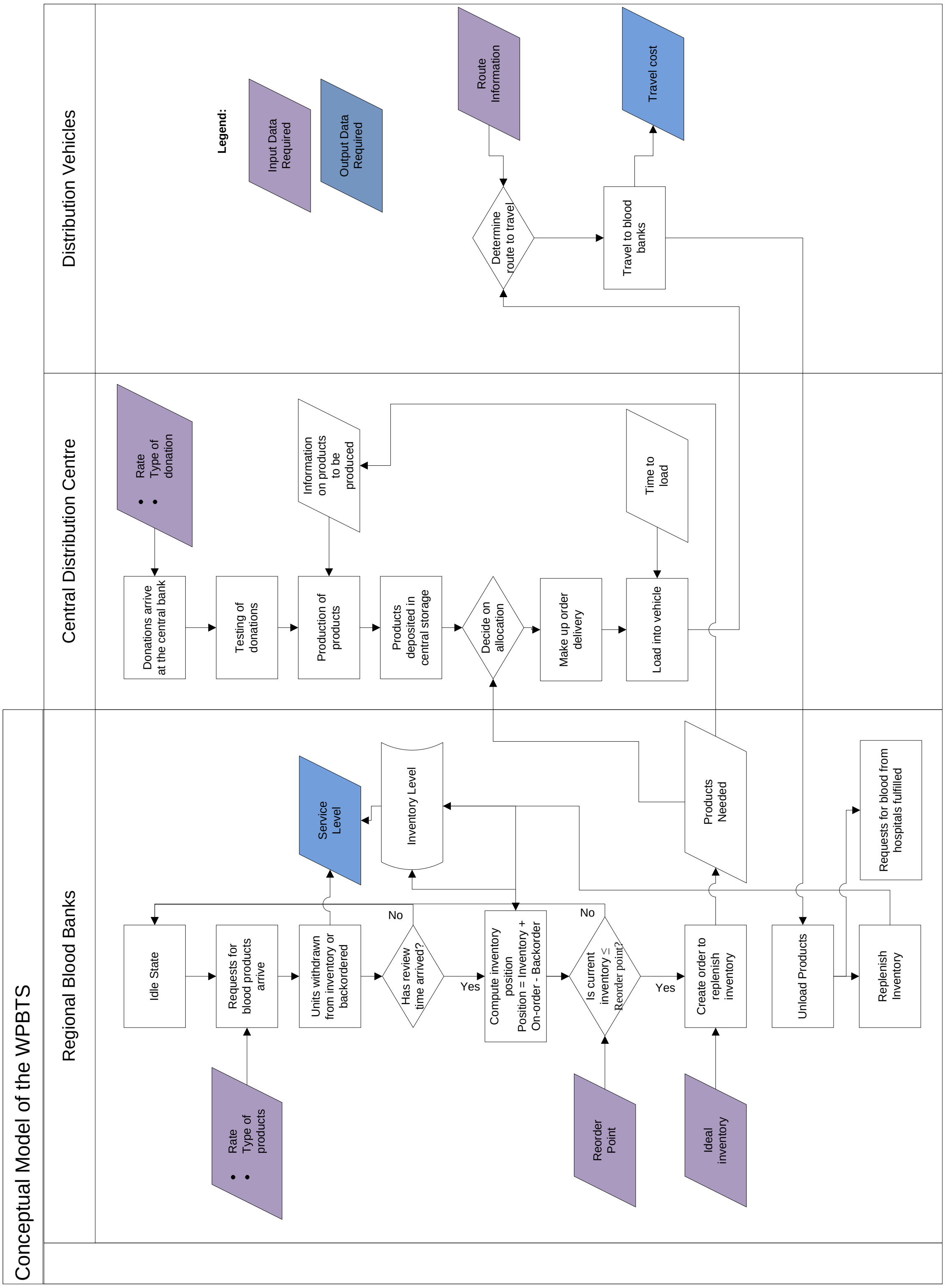

Figure 5.1: Process flow diagram of the WPBTS as conceptual model.

first class was the most important aspect of successfully modelling the system. Sub-classing a server object led to the modelling of the processes at the blood banks, including inventory review, inventory replenishment, creating orders *etc*. Entities that enter and exit the servers were divided into an ordering system (electronic) and a supply system (the physical blood units). The power of the Simio® software is founded in the events that are triggered by these object classes and the steps that model state changes as a response to events (Pegden, 2012). The model of the WPBTS was built on the blueprint of exercises on supply chain modelling in Simio ® by Joines & Roberts (2010). The development of the computerised model started with a simple supply chain with one blood bank and one product to be delivered. This was extended to incorporate a number of products and then to incorporate the correct number of blood banks. Finally the delivery strategy of the WPBTS was modelled using a transporter object and the necessary links. After the computerised model is programmed, validation and verification need to be executed again. Prior to building this model, the student had no experience with the software, and consequently the model was built step by step and the syntax and programming steps were checked as the model grew. The computerised model validation also establishes that the conceptual model was implemented correctly. Law & Kelton (2000) introduce verification and validation factors to ensure model reasonableness. This entails that the behaviour of the model portrays changes made to the model and the input, and were incorporated as follows:

- **Continuity:** The model shows good continuity as a change in input values is reflected in the output. Ensuring a high initial inventory and replenishing it often led to a high service level. Setting the reorder points high meant deliveries were made more often and this led to an increase in transportation costs.

- **Degeneracy:** The degeneracy of the model was tested during its development as the model was built from the bottom up. *I.e.* a blood bank with a single product was tested before extending it to multiple products. The system was similiarly modelled with one blood bank before extending it.

Removing one of the blood banks led to a higher service level and lower vehicle costs.

- **Absurd conditions:** Verification of the model was done by running the model at high demands, high inventories or low inventories. If high demands were placed on the system, service levels were affected significantly. Orders were placed in queues and the central supply was over utilised. In contrast, absurdly high inventories resulted in a perfect service level.

- **Consistency:** Replications in Simio® showed consistent results for consistent parameters.

The simulation model was further checked for logic by face validation. Du Plessis (2008) introduced a table format that can be used for face validation. The face validation of the WPBTS model is summarised in Table 5.3.

Object Function	**Criteria**	**Validated**
Inventory	Inventory decreases when order is met.	✔
	Reorder is created at time and inventory review decision.	✔
	Size of reorder is reasonable.	✔
	Changes in inventory associated with correct product.	✔
Vehicles	Vehicles travel on predefined networks.	✔
	Vehicles only visit blood banks waiting for orders.	✔
	Vehicle *en route* to George travels during night.	✔
	Vehicle travels shortest route when delivering to a selection of banks.	✔
	Distance travelled is updated after every path.	✔
	Distance travelled is correct reflection of path length.	✔
Orders	Orders travel from relevant blood bank to central distribution and back.	✔
	Order is associated with correct blood product and type.	✔

Continued on next page

Object Function	Criteria	Validated
Input Data	Data on demand at the hospitals is reflected in inventory decrease.	✔
	Data on the initial inventory levels is associated with the correct blood bank, product and blood group.	✔
Output Data	Service level values of blood banks contribute to the overall statistic.	✔
	Service levels at blood banks are a reasonable reflection of inventory levels and order sizes.	✔
	The travel distance is a reasonable reflection of the number of visits to blood banks.	✔

Table 5.3: Face validation of the WPBTS simulation model.

5.5.1.3 Data acquisition

In order to enable a study on the management of the valuable resource, Chapman (2007) identifies three reasons for the importance of access to key blood supply chain data as being able to understand the supply chain, for the improvement of blood inventory management practice and for necessary performance evaluations.

From the conceptual model, the input data required was defined. This can be summarised as the arrival rate and type of requests for blood products at the blood banks (*demand*), the arrival rate and type of donations at the central distribution bank (*supply*), the reorder points and inventory levels of the different products (*inventory policy*) and finally information on the *routes* travelled.

Data acquired from and made available by the WPBTS was limited to a daily executive summary of inventory levels at the regional blood banks and central distribution centre for four months, January to April 2012. These summaries were used to calculate the size of orders (*requests for blood*) from the blood banks (daily inventory level subtracted from ideal inventory level). The requests for blood (orders) were then sampled from the PERT distribution in Simio $((Min + (4 * Mode) + Max)/6)$ with the calculated value as the mode to emulate the stochastic nature of the real world system.

As mentioned before, no data was available on the *number of donations* and/or blood products produced. The first experiments conducted used a modelled perfect supply source as discussed in Section 5.6.1.1. In an attempt to model a constrained supply source, the data provided on the inventory level at the distribution centre was used to calculate the daily influx of products in the cases where inventory increased from one day to the next. Although this is not ideal since it is possible that some of the donations are not reflected in the inventory level but sent directly to the blood banks, it is the most realistic representation possible with the data available.

The *inventory policy* data forms part of the decision variables to be optimised. In the case of the as-is system experiments, the current ideal inventory levels were used and a number of possible reorder policies were introduced and discussed in Section 5.6.1.1. The latter is due to the fact that most of the decisions made in this regard are done intuitively by the stakeholders and no data is available.

To adequately model the real world system and to provide for the constraints with regard to the *perishable nature* of the product and a space limitation, the upper limits provided to the algorithm were constrained by the monthly turnover of products as calculated from the data. The actual shelf life of red cell concentrate (RCC) is 42 days (Alexander, 2012), so by ensuring that the inventory at a bank will be requested in a simulation month, this leaves a buffer period of 12 days should the demand be less and products need to be exchanged between blood banks. As mentioned before, deliveries of products are in general done by a third party service provider and data on the routes travelled was not available. The distance travelled by vehicles form an integral part of the problem model, and to enable the modelling of the system, distances were obtained in Google Maps® and summarised in Table 5.4. It is assumed that vehicles travel sequentially to the nearest blood bank and when all products have been delivered the vehicle returns to the distribution centre in Pinelands. Vehicles travelling to George do so at night, and for the purpose of the simulation, it is assumed that the vehicle drives directly to George and back.

	DC	GSH	RCX	TBH	MCV	PAARL	WTR	GEO
DC	0	7.7	5.8	14.6	43	59.6	110	429
GSH	7.7	0	4.3	17.6	41.5	61.6	111	
RCX	5.8	7.7	0	15.5	41.2	59.6	109	
TBH	14.6			0	38.7	51	93.8	
MCV	43				0	56.2	101	
PAARL	59.2					0	57.6	
WTR	110						0	
GEO	429							0

Table 5.4: From/To table of WPBTS delivery routes (km).

5.5.2 Optimisation model of the WPBTS case study

The development and basic concepts of the cross-entropy method (CEM) and the use of the method for MOO were discussed in Chapter 3 and applied to the vehicle routing problem with soft time windows (VRPSTW) in Chapter 4. The VRPSTW called for an associated estimation problem derived from Markov chain principles, with the transition probability matrix updated according to the cross-entropy method to obtain an optimum result. The inventory problem of the WPBTS simplifies to an allocation problem where the inventory levels and re-order points of the different products at the different blood banks need to be assigned, amounting to 194 decision variables as explained in Section 5.3. The number of variables is quite extensive and serves as another challenge for the cross-entropy method for multi-objective optimisation. The algorithm is given limits on which to search for decision variables, with some sets as large as 80. These limits are documented in Appendix B. The combinatorial nature of the problem results in a staggering 4.453×10^{256} of possible solutions to be evaluated, and obtaining a good approximation front is a feat in itself. The objectives to be optimised are the sum of the perceived service levels at the blood banks (to be maximised) against the travelling distance (to be minimised), modelled as the number of kilometers the vehicles travelled between the distribution centre and the blood banks, or against the travelling cost (to be minimised). Table 5.5 lists the symbols used in the solution model of the WPBTS case study.

The service levels are calculated by tallying the inventory available of the

Symbol	**Description**
SL	Total Service Level
TC	Travel Cost
TD	Travelled Distance
i	Blood bank enumerator
j	Blood product enumerator
sl_i	Average service level at blood bank i
$\mathcal{N}_i$	Number of products at blood bank i
$\mathcal{O}_j$	Size of order of product j
$\mathcal{I}_j$	Current inventory level of product j
c_1	Transport cost per unit per kilometer
c_2	Vehicle cost per kilometer travelled

Table 5.5: Symbols used in the solution model of the WPBTS case study.

specified product divided by the size of the order,

$$sl_i = \text{Maximise } \frac{\sum_{j=1}^{\mathcal{N}_i} \min\left(1, \frac{\mathcal{I}_j}{\mathcal{O}_j}\right)}{\mathcal{N}_i} \quad \forall \quad \text{Orders during simulation time},$$

$$SL = \sum_{i=1}^{7} sl_i.$$

Due to the difference in demand and supply of products SL varies considerably per product. Taking the average of this value over all the demanded products provides a single figure per blood bank and assumes equal contributions per product to the service level. This can also be determined by a decision maker, *i.e.* to what extent one product's service level contributes to that of the entire blood bank.

With these objectives in mind, the cross-entropy method is applied to the discrete problem. Decision variables are sampled from truncated Poisson distributions, with parameter λ_x updated at every iteration. The algorithm is given upper and lower bounds for every decision variable which are updated algorithmically.

Algorithm 4 was first introduced by Bekker (2012) and adapted for the case study in this research. Step 11 is executed as follows. The solution space is defined by a set of limits that are in general determined by the perishability constraint and an upper constraint on the inventory level at a particular blood bank (Table

Algorithm 4 MOO blood inventory management with the CEM.

1: Let P be a vector of Poisson distribution parameters, N_m the maximum number of loops, τ the maximum number of evaluations per loop and $\boldsymbol{\mu}$ the set of averages of the objectives in the elite array.
2: $L_c \leftarrow 0$
3: **repeat**
4:     $t \leftarrow 0$
5:     Initialise P_t and $\boldsymbol{\mu}_t$
6:     **repeat**
7:         $t \leftarrow t + 1$
8:         Generate population of N solutions using P_{t-1}
9:         Use discrete event simulation to evaluate each vector and return objective function values SL and TC or TD.
10:         Rank the solutions and the current **Elite** using the threshold $t_h = 0$ to determine new **Elite**.
11:         Update elements in P_t with (3.6)
12:         Smooth $P_t \leftarrow \alpha P_t + (1 - \alpha) P_{t-1}$
13:     **until** $|\boldsymbol{\mu}_t - \boldsymbol{\mu}_{t-1}| \leq \epsilon$ or $t \geq \tau$
14:     $L_c \leftarrow L_c + 1$
15: **until** $L_c > N_m$
16: **Return** elite set

B.1). The solution space is characterised by parameter value λ_x, the mean of a population of x values. Step 11 is consequently executed by dividing the solution space defined by a set of limits into equally sized distributions. These distributions are characterised by a temporary parameter value λ'_{xi} where i is an enumerator for every distribution on variable x. The elite is evaluated and a histogram of the frequency of occurrence of elite values in a particular distribution i is calculated. The limits of the current elite is also used in the next iteration as limits to divide the solution into i distributions. Then (3.6) is used to determine new values for λ'_{xi} from which the next iteration of decision variables is sampled. In addition the frequency of the occurrence of elite values on a distribution i, induces a higher percentage of values sampled in the next iteration (Step 8).

5.5.3 Integrating the optimisation and simulation models

As explained in the previous sections, the simulation model was built in Simio® and the optimisation algorithm coded in Matlab®. A dynamic loaded library is made available as part of the Simio® software and due to its object-oriented nature, objects can be accessed in the library. This dynamic loaded library can be called from Matlab® using the .Net interface, where different modules of Simio®, such as the project, model and experiment interfaces are exposed. This enables a call to the simulation project and, amongst others, a method to run and reset the experiment from Matlab®. Simio® writes results to a text file which is read by Matlab®. A simplified explanation of the integration is displayed in Figure 5.2. To accomplish this, the Simio® API module was used on a .Net interface.

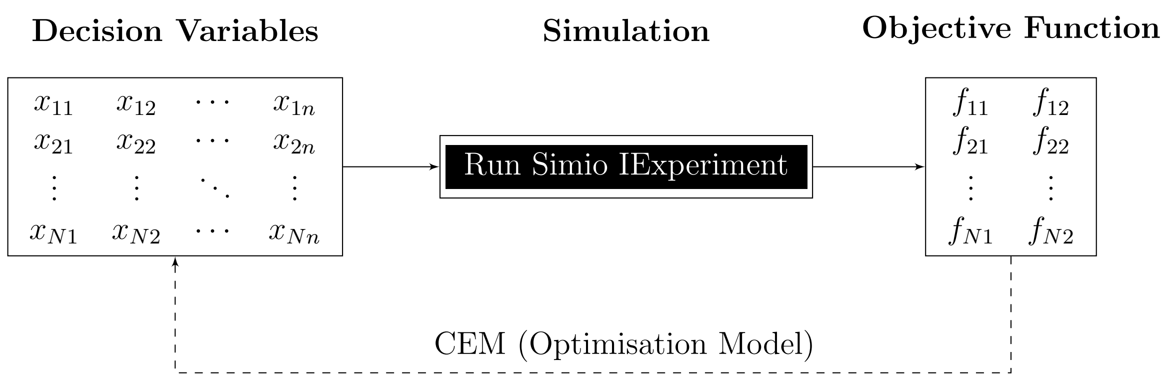

Figure 5.2: Integrating the optimisation and simulation model.

5.6 Experiments and results

The development of the research problem and of the optimisation model was documented in the previous sections. The inventory policy of the WPBTS is investigated by simulating four months of operations of which data was available. The CEM is then used to optimise variables. Experiments are divided into two subgroups, both with the service level (SL) as objective and the first group with travel distance (TD) as additional objective and the second with the travel cost (TC) as objective. For both groups the experimental procedure is explained before results are presented and discussed.

5.6.1 Experiments on the travel distance

The first experiment group is done with the total distance travelled (km) by delivery vehicles in the system (TD) and the service level (SL) as objectives.

5.6.1.1 Experimental procedure

The first results are obtained by simulating the as-is system; *i.e.* using the inventory levels of the WPBTS currently employed. Due to the fact that staff rely on their expertise to judge the reorder point, the as-is reorder points are not obtainable. Experiments are conducted with the reorder point at set percentages of the current inventory levels. The 0.50 reorder percentile of the O^+ LDRC at Groote Schuur Hospital with a current inventory level of 71 would for instance be 35 (values are rounded down). Experiments were conducted in Simio® with a confidence level of 95% and 10 replications executed.

The second experimental procedure uses the CEM to optimise the objective function values of a system with no limit on the supply. This is modelled by introducing a perfect supply source to the central distribution bank. *I.e.* if blood products are requested, it is replenished with a lead time of 0.5 days. Although this is not an accurate representation of reality, the assumption was necessary due to the lack of data and feasible due to the assurance from the WPBTS staff that the donations are generally sufficient for the needs of the province. This experiment uses the integrated optimisation and simulation model (Section 5.5.3). Parameters are set to $\alpha = 0.7$, $N_m = 5$, $\tau = 5$, $N = 50$ and the number of simulation replications to five.

The final experiments in this section derive general supply figures from the central supply centre's data. Data given on the inventory level at the central distribution centre is used to model the influx of blood products into the system by documenting increases in the inventory over the four simulated months. This data on the supply from donations is not ideal, but demonstrates the effect of a limited supply on the rest of the system. Experiments are also conducted with the constrained supply at 125% of derived data and at 75 %. The experimental procedure is similiar to that of the previous experiment with the same parameter setting.

The simulation package provides OptQuest®, an add-in optimisation package. The simulation model is optimised using this package and by providing the same limits to the variables as is used by the CEM-optimisation model. The last results documented is an example of an instance of comparison between the two optimisation techniques. Although providing space for multiple objectives (responses), OptQuest® requires that one of the objectives be named the primary objective. SL was defined as the primary objective as a way to ensure the feasibility of the solution. The add-in generated 250 scenarios of decision variables for comparison.

In summary the experiments of this section:

1. Use the CEM for MOO to optimise the WPBTS system with a perfect supply source.

2. Use the CEM for MOO to optimise the WPBTS system with a constrained supply source.

3. Use the CEM for MOO to optimise the WPBTS system at 125% and 75% of the constrained supply source.

4. Use the OptQuest ®package to optimise the WPBTS system with a perfect supply source.

5. Simulate the as-is system of experiment 1 – 3 at four different reorder levels, 50%, 75%, 90% and 100% of the ideal inventory level.

5.6.1.2 Results

Experiments are run as explained in Section 5.6.1.1 to obtain results. This subsection provides a summary of the results obtained. Experiment 5 generates values for the as-is system that are used as a reference in the other experiments. The as-is values displayed in the figures show the results if the model is run with the inventory policy as currently employed by the WPBTS. The as-is system does not have predefined reorder points, but these are determined by the decision maker at the central distribution centre. To adequately display possible policies, the as-is plot consists of four plots which is differentiated by setting the reorder point to a set percentage of the ideal inventory, 50%, 75%, 90% and 100%.

Experiment 1 – WPBTS system with perfect supply source

Figure 5.3 provides the graphic results of the initial model with a perfect supply source. The figure displays plots for the archive of solutions that were generated and evaluated during the run of the cross-entropy algorithm and the final approximation front that was extracted from these solutions is indicated. In Figure 5.3 the entire solution space is shown and it can be seen that the MOO CEM provided a number of solutions in a specific region. The solutions seem to be restricted to a certain TD level, this is due to the length of simulated time, the number of kilometers that can be completed in a given time period has an upper limit. To provide some clarity, Figure 5.3 is presented again in Figure 5.4 on a different scale, showing only results of $SL \geq 2.5$. This figure shows that the MOO CEM provides a good approximation front of solutions that maintain a higher service level at a lower travel cost, when compared to the current inventory policy. Due to the importance of blood supply to the population of the region, a decision maker will typically determine a minimum service level and choose the solution that will result in the lowest vehicle cost.

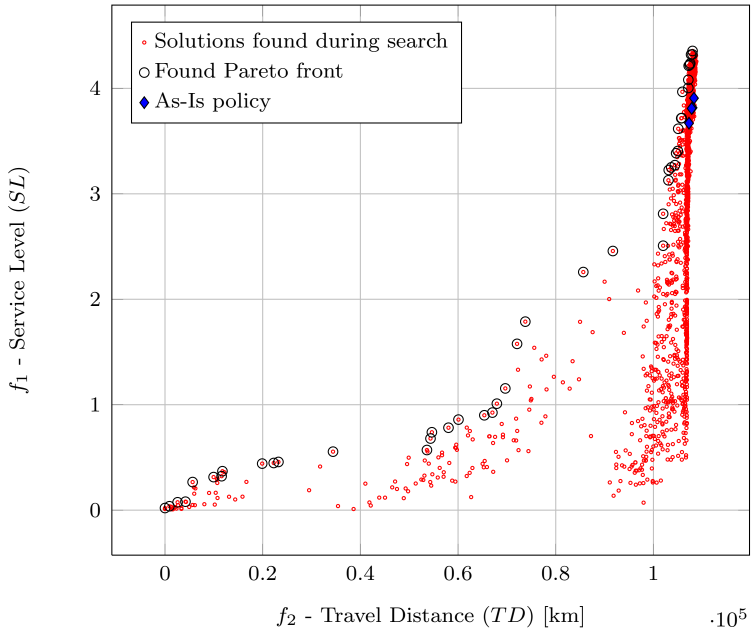

Figure 5.3: Graphic results for system with perfect supply source.

Figures 5.5 and 5.6 show the convergence of the CEM algorithm in the case of the WPBTS model with the perfect supply source. Figure 5.5 shows the movement of λ for two arbitrarily chosen decision variables, x_{73} and x_{74}, or the ideal inventory level and the reorder point of O$^+$ LDRC at Vergelegen Mediclinic. Initially the movement of λ is quite large as it is adjusted by the algorithm according to good

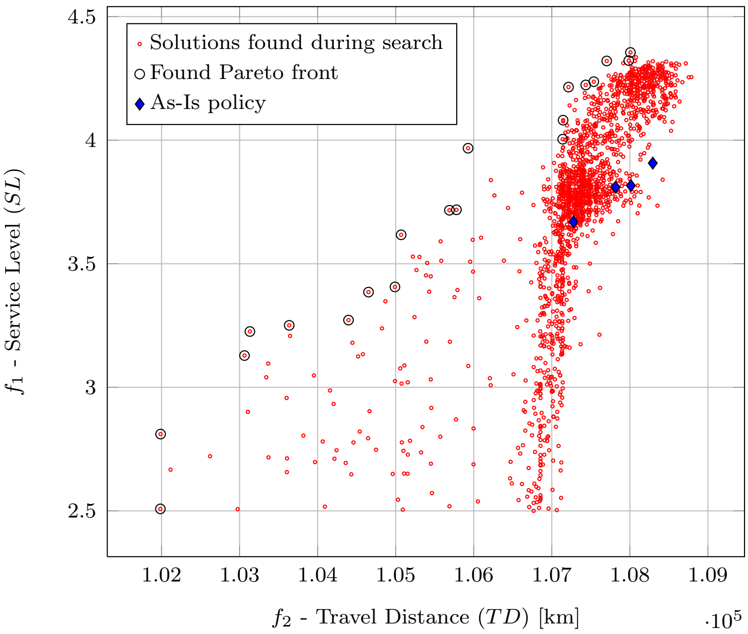

Figure 5.4: Scaled graphic results for system with perfect supply source.

values in the elite set. The size of improvements decreases as the generations progress, indicating that the number of iterations is sufficient. Figure 5.6 displays the values of σ for a number of generations. The value represents the deviation of the λ_x values in the elite, and is an indication of the convergence of the algorithm. The value stabilises relatively quickly, and after the second iteration ($k = 2$) values are generally consistent. These results serve as an indication of the value of the CEM and show that it performs adequately.

Figure 5.7 illustrates the final approximation front of the system model with a perfect supply source. Tables 5.6, 5.7 and 5.8 are included in the document to give an indication of decision variable values for two extreme solutions in the approximation front and one that might be selected by a decision maker. The table provides the ideal inventory level (Inv) and the reorder point (ROP) of a blood group (eg. $\mathbf{O^+}$) and blood product (eg. **WB** – Whole Blood) at a specific blood bank (eg. **RCX** – Red Cross Children's Hospital). It should be noted that despite the fact that solution A and solution B appear to have similiar values for the **TD** objective, solution B provides a decrease of 2000 km in the travel distance which can become more significant given the current trend of the fuel price.

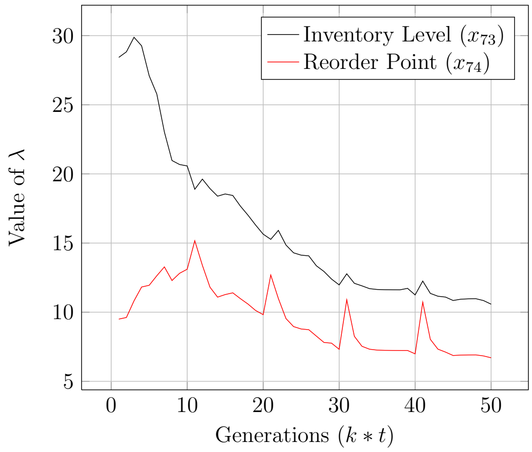

Figure 5.5: Movement of λ for x_{73} and x_{74} in system with perfect supply source, SL vs TD.

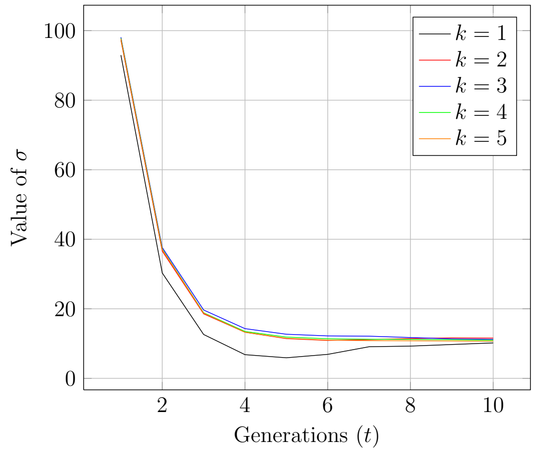

Figure 5.6: Movement of σ for x_{73} in system with perfect supply source, SL vs TD.

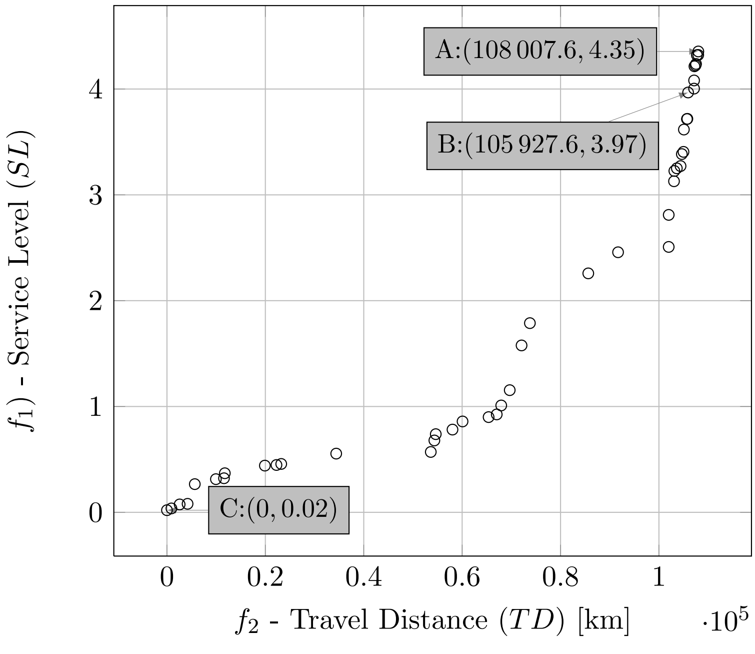

Figure 5.7: Final approximation front for system with perfect supply source.

Although the solution in Table 5.8 is included for the purpose of the study on the cross-entropy method it is of little value for the WPBTS. Standards in health care need to be at a high level to ensure safe healthcare for all. The decision maker in this case will typically decide on a minimum service level which must

		O⁺		**O⁻**		**A⁺**		**A⁻**		**B⁺**		**B⁻**		**AB⁺**		**AB⁻**	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			14	13												
	GSH	33	19	13	10												
	MCV																
	PAARL																
	RCX	4	0	1	1	0	0	13	13	0	0			12	12		
	TBH	17	15	16	10												
	WTR																
RCC	**GEO**	119	115	34	27	105	79	17	16	67	57	7	7	22	14		
	GSH	124	115	20	13	111	109	5	5	66	41	6	4	42	29	3	0
	MCV	67	59	19	9	57	54	14	13	31	17	6	5	11	10		
	PAARL	69	57	16	16	58	54	6	4	43	29	2	2	15	12		
	RCX	81	58	18	13	50	46	13	11	49	40	0	0	23	18	2	2
	TBH	143	100	30	30	134	132	20	15	77	73	8	6	39	22	2	0
	WTR	86	79	8	8	63	43	10	6	57	43	0	0	13	12		
LDRC	**GEO**	36	26	8	6	40	33	6	4	10	6						
	GSH	91	86	20	19	67	52	16	13	28	20	10	3				
	MCV	32	24	14	14	37	36	9	5	14	10	6	0				
	PAARL	8	8			7	7			6	6						
	RCX	51	36	36	34	54	48	13	12	34	26	6	3				
	TBH	69	57	18	10	45	32	20	15	37	25	9	6				
	WTR	13	11			11	6										

Table 5.6: Values of variables for solution A (system with perfect supply source, SL vs TD.

		O⁺		**O⁻**		**A⁺**		**A⁻**		**B⁺**		**B⁻**		**AB⁺**		**AB⁻**	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			14	13												
	GSH	18	7	11	9												
	MCV																
	PAARL																
	RCX	3	0	5	0	0	0	11	5	0	0			2	1		
	TBH	7	5	0	0												
	WTR																
RCC	**GEO**	64	60	12	7	58	41	3	3	41	31	9	8	16	2		
	GSH	103	51	20	18	74	43	5	4	43	14	3	0	23	13	1	0
	MCV	43	35	13	10	52	37	8	6	13	7	4	2	12	9		
	PAARL	42	25	11	4	37	34	5	3	17	17	4	4	14	7		
	RCX	48	35	5	1	14	13	3	1	30	20	0	0	13	4	4	2
	TBH	89	74	17	8	119	112	10	6	45	39	4	0	32	11	2	0
	WTR	76	50	7	3	51	22	2	2	34	16	0	0	8	4		
LDRC	**GEO**	26	8	8	0	20	12	0	0	13	0						
	GSH	61	59	9	5	40	30	14	3	7	2	6	1				
	MCV	25	10	6	6	10	8	6	5	1	0	1	0				
	PAARL	2	0			2	2			2	1						
	RCX	24	8	27	12	44	20	9	6	22	15	5	1				
	TBH	16	14	13	7	25	5	5	0	23	15	5	2				
	WTR	6	0			0	0										

Table 5.7: Values of variables for solution B (system with perfect supply source, SL vs TD).

be upheld and select solutions that lie above this minimum on the approximate Pareto front. Following the validation tests, results for variables in Solution C were expected. Almost all the variables were set to zero with the exception of a few exceptionally low inventory levels. Although showing the general trend in decision variables, a situation like this is assumed to be infeasible.

		O^+		O^-		A^+		A^-		B^+		B^-		AB^+		AB^-	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			0	0												
	GSH	0	0	0	0												
	MCV																
	PAARL																
	RCX	0	0	0	0	0	0	2	0	0	0			0	0		
	TBH	0	0	0	0												
	WTR																
RCC	**GEO**	1	0	0	0	0	0	0	0	0	0	0	0	4	0		
	GSH	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0
	MCV	5	0	0	0	0	0	0	0	3	0	0	0	0	0		
	PAARL	0	0	0	0	2	0	0	0	2	0	0	0	0	0		
	RCX	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	TBH	1	0	0	0	2	0	0	0	0	0	0	0	1	0	0	0
	WTR	10	0	0	0	2	0	0	0	0	0	0	0	0	0		
LDRC	**GEO**	0	0	0	0	0	0	0	0	0	0						
	GSH	2	0	0	0	0	0	0	0	0	0	0	0				
	MCV	0	0	0	0	0	0	0	0	0	0	0	0				
	PAARL	0	0			3	0			0	0						
	RCX	2	0	0	0	2	0	0	0	0	0	0	0				
	TBH	0	0	0	0	0	0	0	0	0	0	0	0				
	WTR	0	0			0	0										

Table 5.8: Values of variables for solution C (system with perfect supply source, SL vs TD.

Experiment 2 – WPBTS system with constrained supply source

Figure 5.8 has a similiar format to Figure 5.3 with scaled results shown in Figure 5.9. In this case, the system is modelled with a constrained supply source, derived from the data on the inventory level at the distribution centre as explained in Section 5.6.1.1. The searched solution space covers a large area that did not deliver a solution on the approximation front. The approximation front is once again superior to that of the as-is system. The final approximation front is shown in Figure 5.10 with two example solutions documented in Tables 5.9 and 5.10.

Experiment 3 – WPBTS with constrained supply source at different percentages

Figures 5.11 and 5.12 present results on the modelled system with a constrained supply source at 75% and at 125% (Figures 5.13 and 5.14). The final approximation fronts and an example of a good result are included in Appendix B. Finally,

Figure 5.15 compares the different constrained supply source experiments. The constrained supply has a significant effect on the service level, but shows similiar trends in terms of the travelled distance to obtain these varied service levels. It can be seen that the constrained supply places an upper limit on the service level that can be obtained.

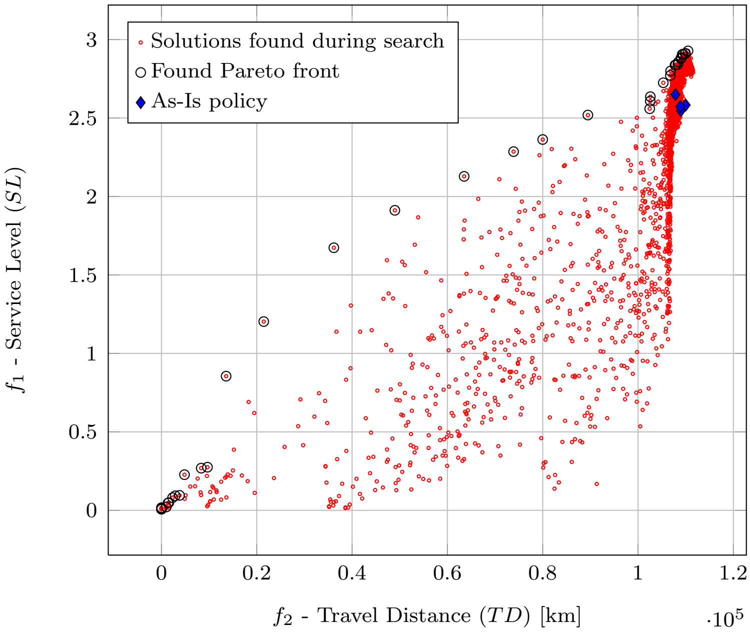

Figure 5.8: Graphic results for system with a modelled constrained supply source.

The tests on the constrained supply source use a relational value, *i.e.* the scenario where one particular blood type is in short supply whilst the others are available in abundance is not tested due to two reasons. The stakeholders confirmed that blood donations were in general sufficient for the needs of the region. The experiments (2 and 3) have the purpose to show the influence of a constrained supply source on the objectives and not to examine the worst case scenario. In addition, the objectives investigated are based on the average over all the blood products at a specific blood bank and further summed to generate a singular objective function value. This confirms the experimentation with the entire system and/or supply. Sufficient data on the donations in the region would eliminate the need for assumptions.

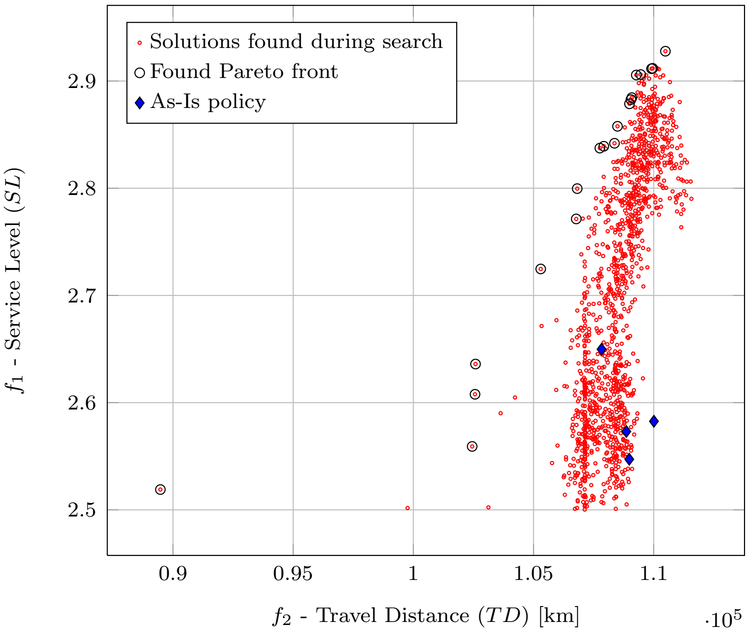

Figure 5.9: Scaled graphic results for system with a modelled constrained supply source.

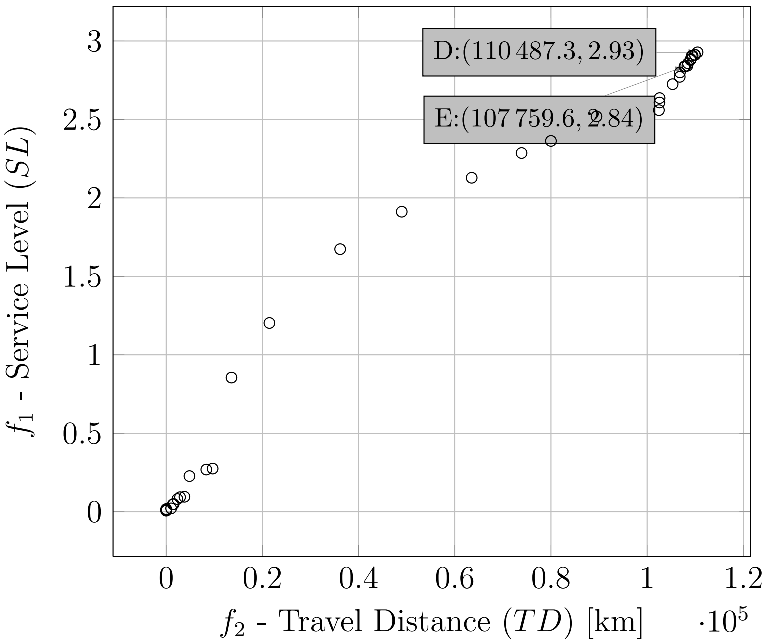

Figure 5.10: Final approximation front for system with a modelled constrained supply source.

		O$^+$		O$^-$		A$^+$		A$^-$		B$^+$		B$^-$		AB$^+$		AB$^-$	
		Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP
WB	GEO			13	6												
	GSH	22	20	12	3												
	MCV																
	PAARL																
	RCX	2	0	1	1	0	0	13	11	0	0			16	15		
	TBH	15	9	18	18												
	WTR																
RCC	GEO	115	86	33	26	104	50	16	12	65	61	10	10	23	15		
	GSH	130	107	26	16	111	42	3	1	44	34	6	1	39	13	1	0
	MCV	84	64	12	9	74	52	15	9	31	19	8	6	16	7		
	PAARL	57	40	17	8	61	42	7	5	38	21	3	3	20	13		
	RCX	62	47	21	10	58	22	10	7	47	36	0	0	13	13	3	3
	TBH	133	128	35	14	125	108	18	14	79	32	8	8	35	12	3	0
	WTR	84	74	7	2	63	45	5	5	53	49	0	0	16	16		
LDRC	GEO	30	24	8	1	38	13	6	2	12	10						
	GSH	75	68	18	10	42	21	15	8	21	12	11	5				
	MCV	34	31	11	4	39	20	8	8	16	7	7	2				
	PAARL	8	5			7	5			6	4						
	RCX	35	29	31	15	53	38	13	11	22	19	4	2				
	TBH	64	42	17	11	38	22	16	9	35	8	8	4				
	WTR	9	6			11	9										

Table 5.9: Values of variables for solution D (system with modelled constrained supply source).

		O$^+$		O$^-$		A$^+$		A$^-$		B$^+$		B$^-$		AB$^+$		AB$^-$	
		Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP	Inv	ROP
WB	GEO			5	4												
	GSH	24	22	10	2												
	MCV																
	PAARL																
	RCX	1	0	1	0	0	0	15	9	0	0			11	8		
	TBH	14	1	19	10												
	WTR																
RCC	GEO	112	88	32	15	82	28	6	5	60	57	9	6	20	14		
	GSH	130	107	26	16	111	42	3	1	44	34	6	1	39	13	1	0
	MCV	74	57	16	3	75	45	18	12	25	25	4	4	8	2		
	PAARL	70	55	15	8	38	23	6	4	34	12	7	3	18	11		
	RCX	60	15	13	5	60	20	2	2	43	41	0	0	8	7	7	7
	TBH	119	104	31	12	83	81	18	18	36	21	7	3	36	9	0	0
	WTR	85	81	3	2	65	36	10	4	52	19	0	0	16	10		
LDRC	GEO	28	28	8	1	35	13	7	0	8	0						
	GSH	85	66	10	9	37	19	11	7	23	11	10	3				
	MCV	35	19	8	2	24	15	5	5	8	7	6	3				
	PAARL	9	4			8	0			2	2						
	RCX	20	12	32	16	36	29	14	8	26	19	0	0				
	TBH	34	16	11	6	32	15	9	8	30	5	5	2				
	WTR	4	2			10	8										

Table 5.10: Values of variables for solution E (system with modelled constrained supply source).

Experiment 4 – Generating values through OptQuest

The final results in this section provide an example of a comparison in the

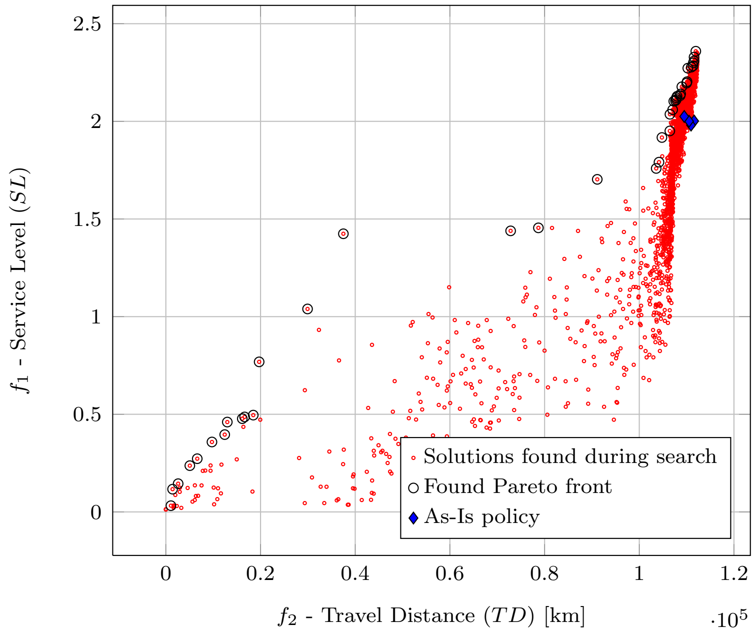

Figure 5.11: Graphic results for system with a modelled constrained supply source at 75%.

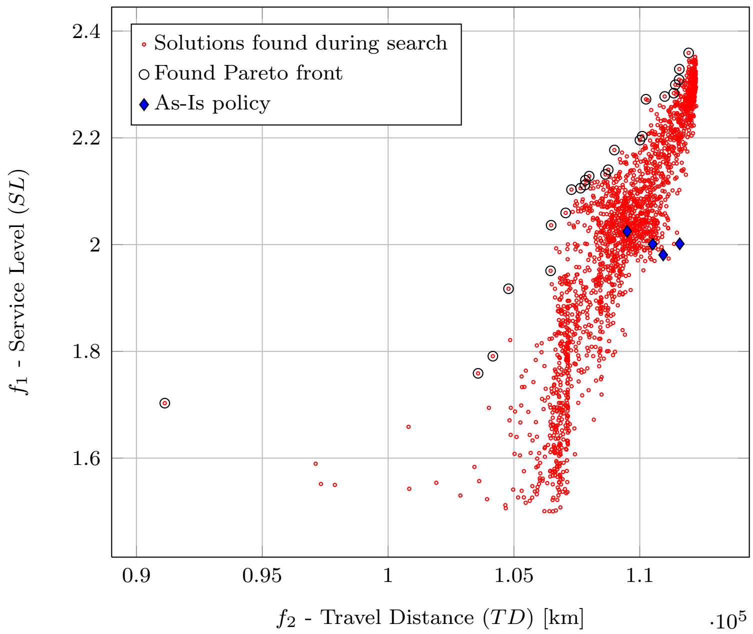

Figure 5.12: Scaled graphic results for system with a modelled constrained supply source at 75%.

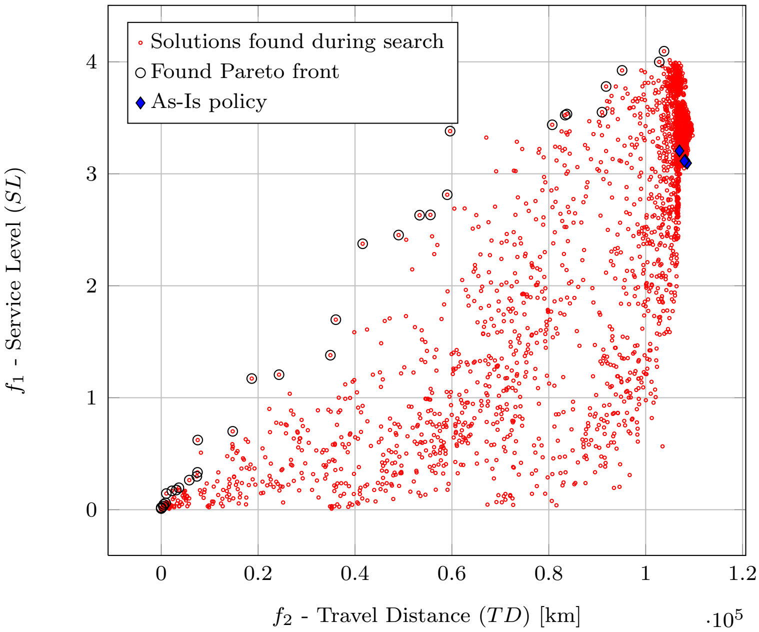

Figure 5.13: Graphic results for system with a modelled constrained supply source at 125%.

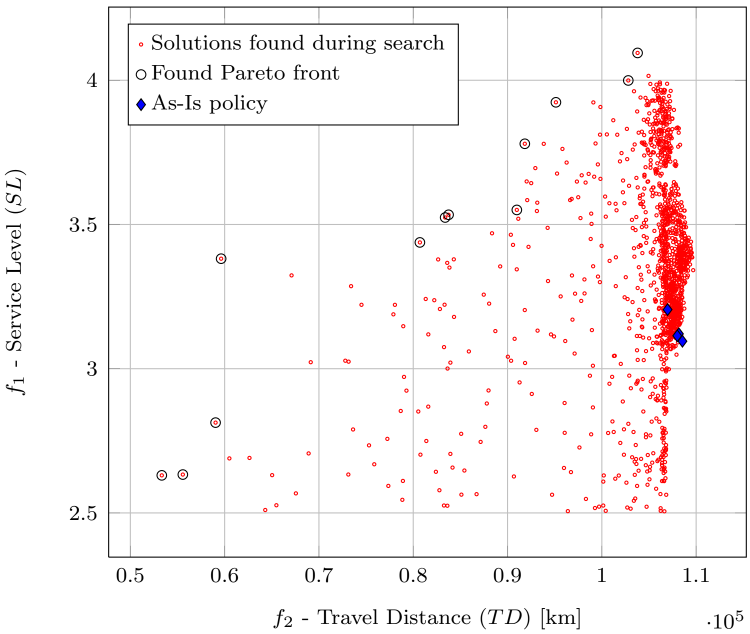

Figure 5.14: Scaled graphic results for system with a modelled constrained supply source at 125%.

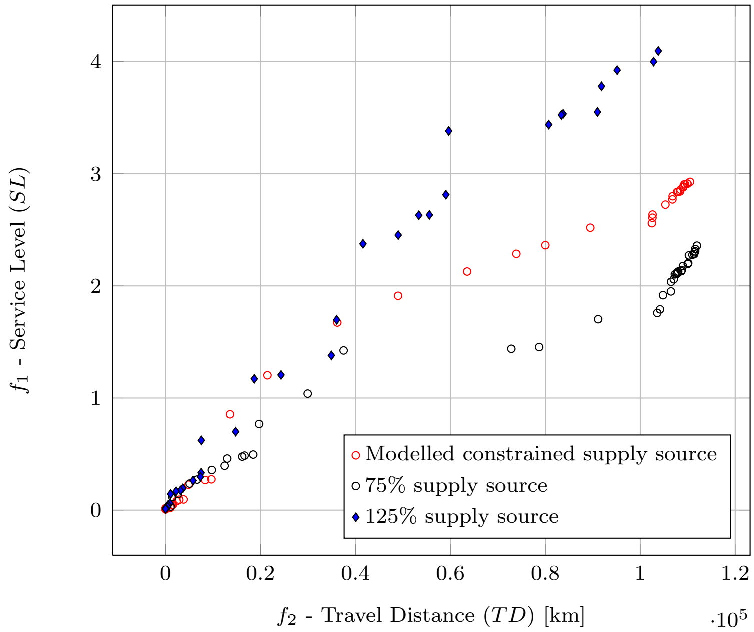

Figure 5.15: Comparison of final approximation fronts with varying limitations on the supply source.

performance of the CEM algorithm and the OptQuest®package. A scientific study where an extensive comparison of the methods are done with different random numbers was deemed infeasible due to time considerations, as one instance of the CEM algorithm is generated in about 36 hours of computational time. The experiment is however still included (Figure 5.16) to illustrate how such a comparison would be approached and to show general results of a comparison. The hyperarea (I_H) was calculated with a reference point ($108\,007.64, 0.02$) for both approximation fronts. The CEM solution generated a front with $I_H = 120\,288.85$ and the OptQuest package a front with $I_H = 6\,495.21$. The cross-entropy method is superior in both diversity and in the objectives obtained, although the OptQuest package provides good solutions that are comparable to the CEM solutions in the area that was searched. The OptQuest package has a limited range of solutions searched, but this could also be an effect of setting the service level (SL) as the primary objective. The solutions provided by the CEM algorithm are in general more diverse and non-dominated if compared to the solutions generated by OptQuest.

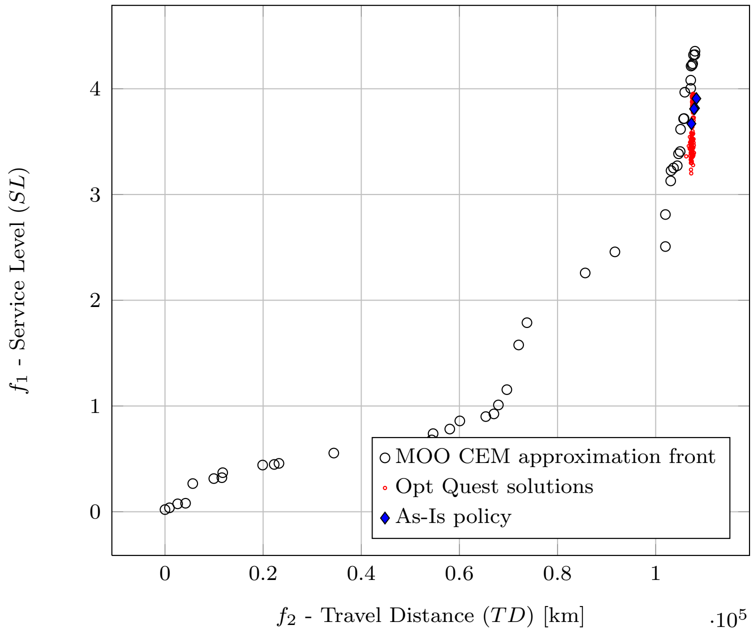

Figure 5.16: Comparison of OptQuest add-in package to CEM algorithm for system with a perfect supply source.

General comments on decision variables

Figures 5.17 and 5.18 are included in an attempt to provide a generalization of decision variables selected to aid decision makers. The final values for λ as calculated by the CEM algorithm are shown for variables representing the ideal inventory levels (Figure 5.17) and the suggested reorder points (Figure 5.18). In general, determined inventory levels were lower than that of the as-is system in the case of large values and higher in the case of small values. The 75% constrained supply source shows values that are all relatively low and with little deviation in the inventory levels of different problems. In the case of the reorder points (Figure 5.18), the values for both the perfect supply source and 125% constrained supply source are in general higher than that of the 100% constrained source which in turn is higher than the 75% constrained supply source. This trend can also be used in the case of dynamically user-determined reorder point values. When supply to the system is low, reorder points could for example be lowered simultaneously. These statements are made in general as there are some deviations as can be seen in the figures. The difficulty to generalise results once again emphasises the complexity of the problem and the worth of the cross-entropy method.

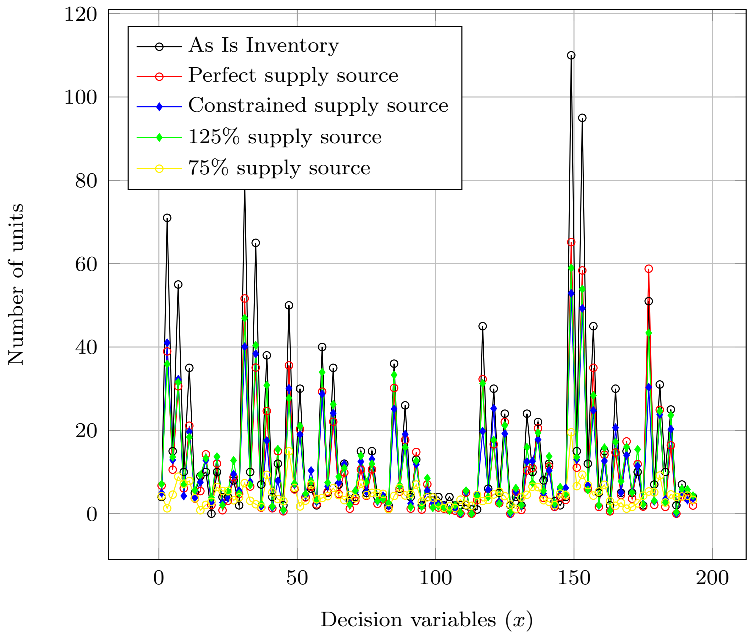

Figure 5.17: Comparison of final values for λ_x of inventory levels for experimental instances, SL vs TD.

5.6.2 Experiments on the travel cost

The second experiment group is done with the total travel cost (TC) as explained in Section 5.5.2 and the service level (SL) as objectives. While the previous experiments with the travel distance (TD) could be used in the case of a WPBTS vehicle that only carries the blood products, this section provides results in the case of a dedicated courier. While the TD is not influenced by the number of products that need to be delivered, this will have an effect on objective TC as a third party service provider will in all likelihood charge more for the amount of the capacity of the vehicle used. To adequately investigate this, a cost is assigned per unit of blood per kilometer travelled (c_1) in addition to a set cost per kilometer travelled (c_2).

5.6.2.1 Experimental procedure

The value TC is computed with two variables c_1 (cost per unit of blood per kilometer transported) and c_2 (cost per kilometer travelled) which make up the

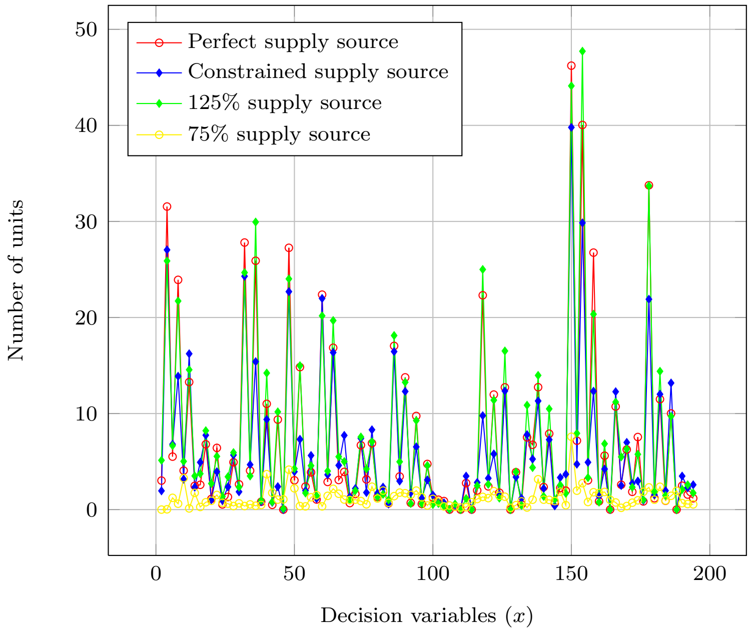

Figure 5.18: Comparison of final values for λ_x for reorder points of experimental instances, SL vs TD.

costing structure of a third party service provider or courier. With no data available on the system costs, experiments are conducted on the WPBTS system with a modelled perfect supply source at four different combinations for c_1 and c_2 to determine the effect of the costing structure on the results. The different combinations of values for c_1 and c_2 are illustrated in Figure 5.19. The system is once again optimised with the cross-entropy method, and results are shown in a similiar fashion to that of Section 5.6.1 with an as-is system modelled at different reorder percentages, the solution space searched and finally isolating the approximation front of results.

The system is then modelled with a constrained supply source with values $c_1 = 0.01$ and $c_2 = 0.5$ to show the effect of a limited supply on the objectives. Finally the OptQuest package is again used to generate 250 solutions and the final results compare the CEM performance to that of OptQuest. All CEM experiment parameters are set to $\alpha = 0.7$, $N_m = 5$, $\tau = 5$, $N = 50$ and the number of replications to five. The OptQuest experiment was run at a confidence level of 0.9, 250 scenarios were generated and properties were given the same limits as

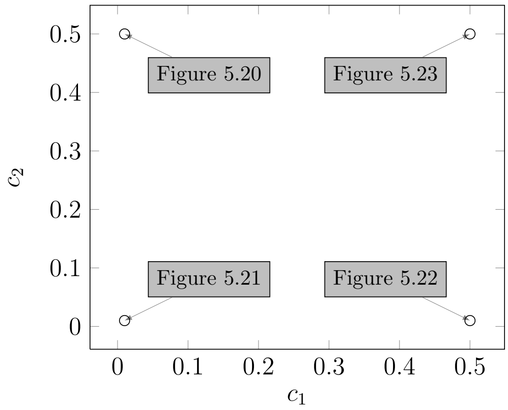

Figure 5.19: Different combinations of costs used in experiments.

that of the CEM experiment.

1. Use the CEM for MOO to optimise the WPBTS system with a perfect supply source at different costing structures.

2. Use the CEM for MOO to optimise the WPBTS system with a constrained supply source.

3. Use the OptQuest ®package to optimise the WPBTS system with a perfect supply source.

4. Simulate the as-is system of all instances of experiments 1 and 2 at four different reorder levels, 50%, 75%, 90% and 100% of the ideal inventory level.

5.6.2.2 Results

Results are presented in a similiar fashion as in Section 5.6.1.2 with figures showing a combination of the search space of the solutions, the approximation front found by the cross-entropy algorithm and the as-is solutions generated with different reorder point percentages (Experiment 4).

Experiment 1 – Different cost structures of the WPBTS system with a perfect supply source

Four experiments were conducted with different combinations of c_1 and c_2 to illustrate the effect of the costing structure (refer to Figure 5.19). Figure 5.20

shows results for the process with costing structure $c_1 = 0.01$ and $c_2 = 0.5$, Figure 5.21 for process with costing structure $c_1 = 0.01$ and $c_2 = 0.01$, Figure 5.22 for process with costing structure $c_1 = 0.5$ and $c_2 = 0.01$ and finally Figure 5.23 for process with costing structure $c_1 = 0.5$ and $c_2 = 0.5$. Final approximation fronts and selected solutions are included in Appendix B.

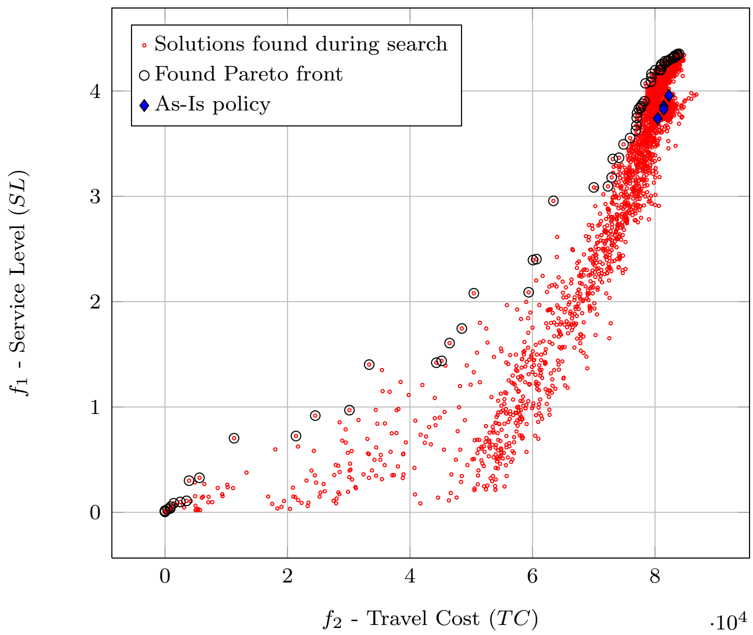

Figure 5.20: Graphic results for system with a perfect supply source ($c_1 = 0.01, c_2 = 0.5$).

Figure 5.24 shows the effect of the costing structure on the final approximation fronts of the different experiments. The solution is more sensitive to an increase in c_1 than that of c_2, giving an indication of an ideal costing structure (the lower c_1 is the better). The service levels of the four different experiments show the same trend and values, proving the service level to be fairly independent of the costing structure.

Experiment 2 – WPBTS system with a constrained supply source

The effect of a constrained blood supply is shown in Figure 5.25. Here the cross-entropy method found solutions at a higher service level but close to the TC values of the modelled as-is system. There are however some solutions that have a lower

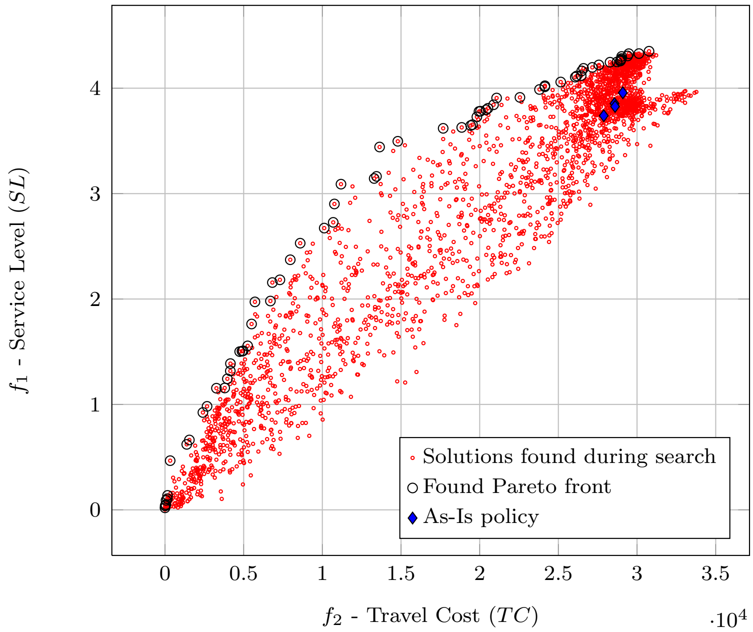

Figure 5.21: Graphic results for system with a perfect supply source ($c_1 = 0.01, c_2 = 0.01$).

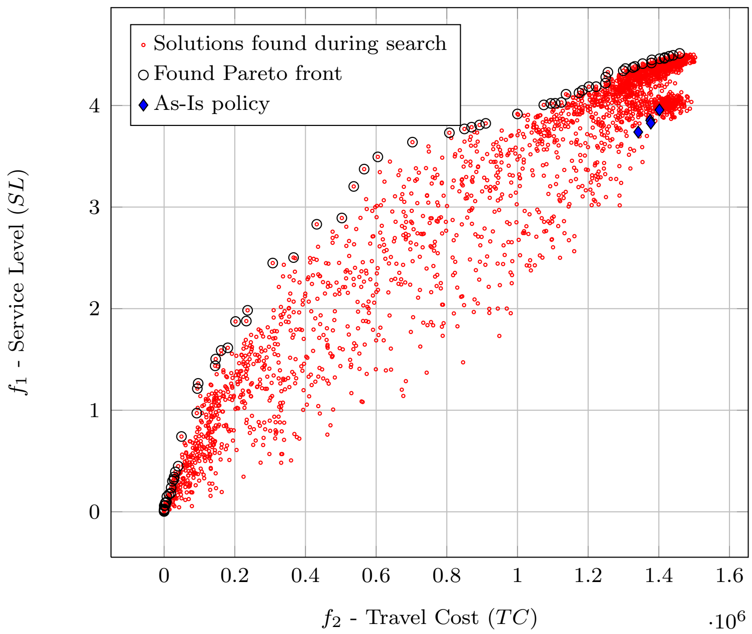

Figure 5.22: Graphic results for system with a perfect supply source ($c_1 = 0.5, c_2 = 0.01$).

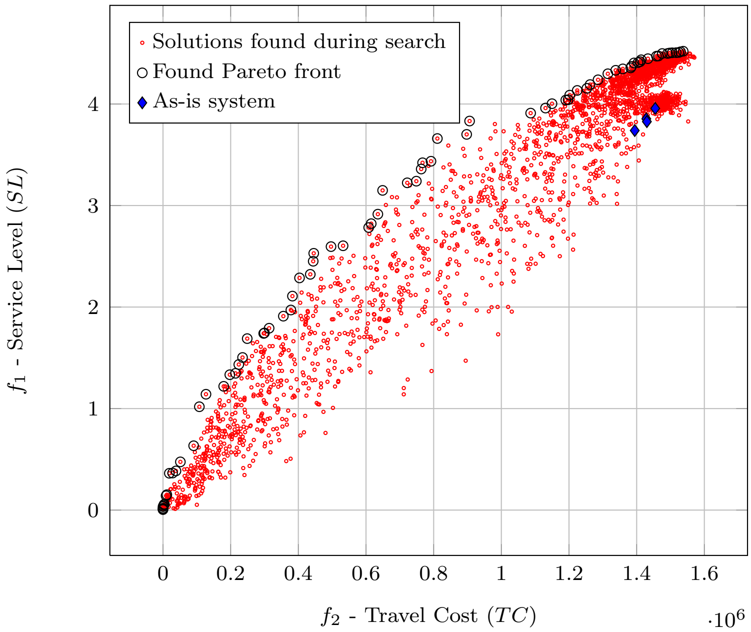

Figure 5.23: Graphic results for system with a perfect supply source ($c_1 = 0.5, c_2 = 0.5$).

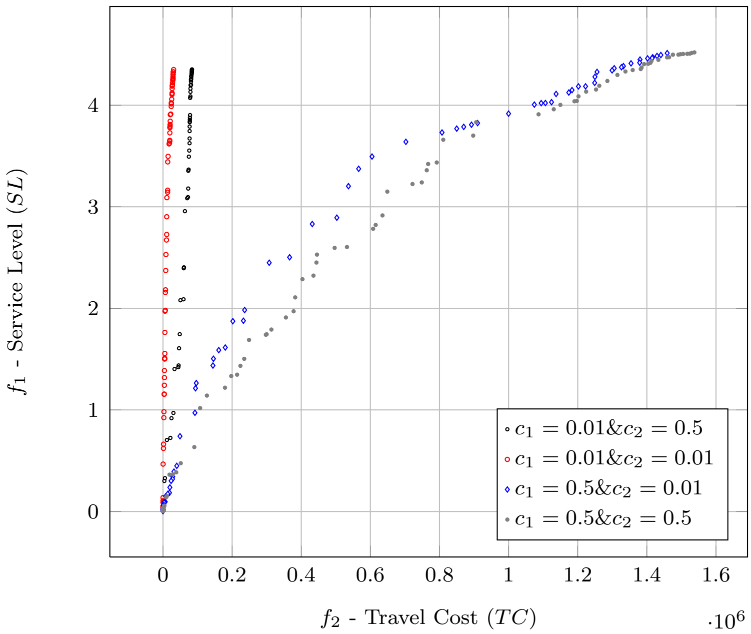

Figure 5.24: Graphic results for system with a perfect supply source with different costing structures.

travel cost at the same service level as the as-is system. The final approximation front is included in Appendix B.

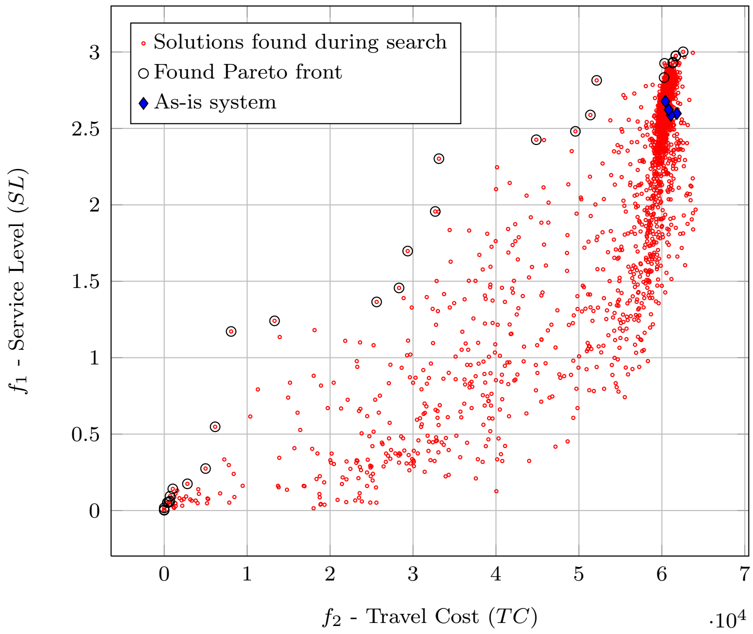

Figure 5.25: Graphic results for system with a modelled constrained supply source.

The last result in this section shows the comparison of the CEM for MOO to that of the OptQuest add-in. As mentioned in the travelled distance experiment (Section 5.6.1.2), this is merely one instance of comparison to give an idea of the difference in output of the two optimisation techniques. The hyperarea (I_H) was calculated from a reference point (83 948.91, 0.0043) for both approximation fronts. The CEM solution generated a front with I_H = 144 461.99 and the OptQuest package a front with I_H = 77 250.07. Once again the cross-entropy method is superior in both diversity and in the objectives obtained, although the OptQuest package provides good solutions that are comparable to the CEM solutions in the area that was searched. The main reason for the lack of diversity could be ascribed to the definition of the SL as primary objective in the case of the OptQuest solution.

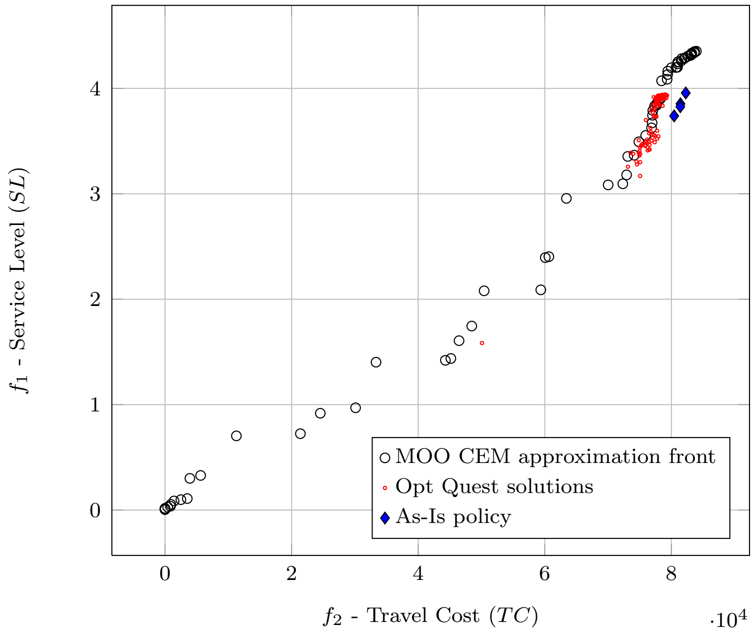

Figure 5.26: Comparison instance of OptQuest add-in package to CEM algorithm for modelled perfect supply source ($c_1 = 0.01, c_2 = 0.5$).

5.7 Concluding remarks on Chapter 5

The aim of this chapter was to model a real world problem and apply the CEM for MOO to the problem. A case study was done on the supply chain and inventory management of the WPBTS. A simulation model and optimisation model were built and integrated to provide results on the inventory policy of the system. It was found that the CEM provides an approximate front of solutions that supply the decision maker with a number of alternatives that will lead to an improvement of the current system. In addition, the experiments showed that the CEM can successfully be applied to a Simio® simulation model with a large number of decision variables and large ranges.

Future work include the memory management of the API programming, which is possibly a Simio® software problem. The blood supply chain problem lends itself to a number of available studies, such as optimising the blood platelet policy, applying the same study to the SANBS and improving the accuracy of the simulation model by also modelling the processing steps at the central distribution centre, the inclusion of actual data on the supply of blood and the modelling of demand at the lowest level of the hospital.

The application of the CEM to a second combinatorial problem was presented in this chapter as a second case study in this research. The next chapter concludes this document with a summary of the research and suggestions for future work.

CHAPTER 6

RESEARCH SUMMARY AND CONCLUSIONS

The research conducted was presented in the previous chapters. This chapter serves as a summary of the findings of the study, suggestions for future work and finally a summary of the contribution to the scholarly field.

6.1 Project summary

The research aim of the project was introduced in **Chapter 1** and can be summarised as applying the cross-entropy method (CEM) for multi-objective optimisation (MOO) to combinatorial problems. To achieve this, a number of research objectives were defined and accomplished during the research. In **Chapter 2**, the field of MOO was introduced through a condensed overview of the literature and basic concepts. After this the CEM was presented with a particular focus on the use of the method for combinatorial optimisation. **Chapter 3** introduced the CEM for MOO as the focal point of the research. This method was subsequently applied to two problems, the discrete multi-objective vehicle routing problem with soft time windows (VRPSTW) in **Chapter 4**, and finally a case study in blood inventory management at the Western Province Blood Transfusion Service in **Chapter 5**. The application of this method was achieved with reasonable success and the overall aim of the research was accomplished. Finding an approximate shape for the Pareto front enables the decision maker to better understand the relationship among objectives, which means that exact solutions are thus not always necessary. In both cases the CEM proved that it is a worthy technique in the MOO of combinatorial problems. Good approximation fronts were obtained with relatively few iterations.

Both application areas were first investigated by doing a literature study on the relevant fields. In the VRPSTW case, model formulation was predominantly done mathematically. The associated estimation problem was Markov chain inspired, with a transition probability matrix an integral part of the route construction algorithm. The biggest challenge of applying the method to the VRP was the two dimensional solution space and the large number of constraints. The application was however deemed successful as the MOO CEM was applied to a number of recent benchmark problems and results showed a progression toward a Pareto approximation front before parameters stabilised.

In the WPBTS case study, the conceptual model was developed after interviews and observations of the real world system. Simulation was chosen as the appropriate modelling technique due to the general stochasticity, time-related processes and the inter-connectedness of elements in the system. After the simulation language was mastered and the model built, the biggest challenge was to integrate the simulation model in Simio® with the optimisation model in Matlab®. Using the API of the simulation package, this was done sufficiently, although memory management remains a problem. In addition, challenges that are part of doing research in a real world with regard to collaboration with stakeholders and the attainment of data was overcome. The near-optimisation of the inventory policies was conducted through an associated estimation problem using a Poisson distribution of each of the 194 decision variables. The MOO CEM produced results that showed improvement of the real world system is possible. The final steps of the case study consisted of investigating the effect that a constrained supply of blood has on the rest of the distribution service and the effect of a delivery costing structure on the objectives. The interested reader can find both optimisation models (VRPSTW and WPBTS) and the WPBTS simulation model on the accompanying compact disc.

6.2 Suggestions for future research

The project originated in a fairly new research development and subsequently parts of the project can be considered as novel research. During the progression of the project and the research conducted, a number of areas were identified that could present itself as future research topics. These are briefly mentioned and summarised below.

1. One of the major strengths of the CEM lies in its application to stochastic problems. The application to the vehicle routing problem, albeit successful, can be extended to investigate the use of the method to solve the stochastic VRP.

2. At the time of research, results on the VRPSTW benchmark problems proposed by Castro-Gutierrez *et al.* (2011) was not readily available. Should results be made available of studies that apply other methods (*e.g.* the NSGA-II) to the VRPTW, a comprehensive analysis of the performance of the CEM for MOO in vehicle routing can be conducted.

3. As explained in **Chapter 5**, a number of assumptions and simplifications were made in modelling the blood supply chain of the WPBTS. The following items could be the subjects of future research.

   (a) Conduct a similiar study on the inventory and general supply chain management of blood platelets in the context of the WPBTS.

   (b) With the availablity of data on the supply from the donor basis in the Western Province, the study presented here can be adapted to incorporate the actual stochastic supply and the inventory policy can be updated.

   (c) Model the processes at the central blood centre to incorporate the lead time of testing products, the actual number of products that are produced and that are subsequently available to the regional blood banks.

   (d) With the availability of data on the fluctuation in demand at the individual hospitals, the model presented can be extended to incorporate the final tier of the supply chain and the effect this has on the inventory levels and service levels.

(e) Investigate an improved inventory policy with independent service levels at the different blood banks as objectives, *i.e.* seven or eight objectives.

4. A project with a greater scope can examine the inventory and supply policies of the SANBS.

6.3 Value of the study

The work done contributes to the scholarship of three different fields. By achieving the research aim the study provided more evidence of the worth of the CEM for MOO by successfully applying it to combinatorial problems. The work done on the multi-objective VRPSTW introduces a new technique to the field while providing results for the newly proposed benchmark problems. At the time of writing this thesis, no reference solutions were available. The field of blood management benefits from the study as a new technique and problem formulation is introduced, while the case study provided an improved inventory policy to the real world system of the WPBTS.

BIBLIOGRAPHY

Alexander, B. (2012). Personal interview: 2 May 2012. 58, 64, 70, 75

Allen, M. (2001). Component Preparation and Storage. In C. Hillyer, ed., *Handbook of Transfusion Medicine*, Academic Pr. 58, 59

Alvarenga, G., Mateus, G. & De Tomi, G. (2007). A genetic and set partitioning two-phase approach for the vehicle routing problem with time windows. *Computers & Operations Research*, **34**, 1561–1584. 34

Bekker, J. (2012). *Applying the cross-entropy method in multi-objective optimisation of dynamic, stochastic systems*. Ph.D. thesis, Stellenbosch University. 25, 26, 77

Bekker, J. & Aldrich, C. (2010). The cross-entropy method in multi-objective optimisation: An assessment. *European Journal of Operational Research*, **211**, 112–121. 2, 3, 4, 25, 27, 35, 39

Beliën, J. & Forcé, H. (2011). Supply chain management of blood products: a literature review. *European Journal of Operational Research*. 57, 66, 67

Berger, J. & Barkaoui, M. (2003). A hybrid genetic algorithm for the capacitated vehicle routing problem. In *Genetic and Evolutionary Computation - GECCO 2003*, 198–198, Springer. 32

Bräysy, O. & Gendreau, M. (2005). Vehicle routing problem with time windows, part I: Route construction and local search algorithms. *Transportation science*, **39**, 104–118. 32

CASTRO-GUTIERREZ, J., LANDA-SILVA, D. & MORENO PEREZ, J. (2011). Nature of real-world multi-objective vehicle routing with evolutionary algorithms. In *2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 257–264, IEEE. xxi, 34, 37, 38, 42, 43, 47, 55, 105, A-1

CHAPMAN, J. (2007). Unlocking the essentials of effective blood inventory management. *Transfusion*, **47**, 190S–196S. 74

CHAPMAN, J., GARWOOD, P. & KNOWLES, S. (2009). Supply and demand for blood and blood components and stock management. In M. Contreras, ed., *ABC of transfusion*, chap. 2, BMJ Books. 58, 60, 61, 63

CLARKE, G. & WRIGHT, J. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 568–581. 31

COELLO COELLO, C. (2006). Evolutionary multi-objective optimization: a historical view of the field. *Computational Intelligence Magazine, IEEE*, **1**, 28 – 36. 8, 14

COELLO COELLO, C. (2009). Evolutionary multi-objective optimization: some current research trends and topics that remain to be explored. *Frontiers of Computer Science in China*, **3**, 18–30. 2, 9, 11

COELLO COELLO, C.A., LAMONT, G.B. & VAN VELDHUIZEN, D.A. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, 2nd edn. 8, 10, 14, 15, 17

COOPER, W. (2001). Pathwise properties and performance bounds for a perishable inventory system. *Operations Research*, 455–466. 60

CORDEAU, J., LAPORTE, G. & MERCIER, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 928–936. 32

CRAINIC, T. & LAPORTE, G. (1997). Planning models for freight transportation. *European Journal of Operational Research*, **97**, 409–438. 29

D'ACQUISTO, G. & NALDI, M. (2005). Cross-entropy-based adaptive optimization of simulation parameters for Markovian-driven service systems. *Simulation Modelling Practice and Theory*, **13**, 619 – 645. 24

DANTZIG, G. & RAMSER, J. (1959). The truck dispatching problem. *Management science*, 80–91. 29

DE ANGELIS, V., RICCIARDI, N. & STORCHI, G. (2001). Optimizing blood assignment in a donation–transfusion system. *International Transactions in Operational Research*, **8**, 183–192. 60, 61, 62

DE BOER, P.T., KROESE, D., MANNOR, S. & RUBINSTEIN, R. (2005). A tutorial on the cross-entropy method. *Annals of Operations Research*, **134**, 19–67. 20, 23, 24, 35

DEB, K. (2001). *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley. 13, 14

DEB, K., PRATAP, A., AGARWAL, S. & MEYARIVAN, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, **6**, 182–197. 15, 16, 33, 34

DORIGO, M., MANIEZZO, V. & COLORNI, A. (1996). Ant system: optimization by a colony of cooperating agents. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, **26**, 29–41. 19

DU PLESSIS, A. (2008). *Determining Tactical Operations Policies for an auto carrier using discrete-event Simulation*. Master's thesis, Stellenbosch University. 73

DU PREEZ, J. (2011). *A Study of Reconfigurable Manufacturing Systems with Computer Simulation.*. Master's thesis, Stellenbosch University. 25

FONSECA, C. & FLEMING, P. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary computation*, **3**, 1–16. 13, 15

FONSECA, C., FLEMING, P. *et al.* (1993). Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In *Proceedings of the fifth international conference on genetic algorithms*, vol. 1, 416, Citeseer. 12, 15

FONSECA, C., KNOWLES, J., THIELE, L. & ZITZLER, E. (2005). A tutorial on the performance assessment of stochastic multiobjective optimizers. In *Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, vol. 216. 13

GARCIA-NAJERA, A. & BULLINARIA, J. (2011). An improved multi-objective evolutionary algorithm for the vehicle routing problem with time windows. *Computers & Operations Research*, **38**, 287–300. 33, 34

GASPAR-CUNHA, A. & COVAS, J. (2003). A real-world test problem for EMO algorithms. In *Evolutionary Multi-Criterion Optimization*, 66–66, Springer. 25

GEIGER, M.J. (2008). A computational study of genetic crossover operators for multi-objective vehicle routing problem with soft time windows. *CoRR*, **abs/0809.0410**. 33

GENDREAU, M., HERTZ, A. & LAPORTE, G. (1994). A tabu search heuristic for the vehicle routing problem. *Management Science*, 1276–1290. 32

GHOSEIRI, K. & GHANNADPOUR, S. (2010). Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm. *Applied Soft Computing*, **10**, 1096–1107. 33

GIL, C., MÁRQUEZ, A., BAÑOS, R., MONTOYA, M. & GÓMEZ, J. (2007). A hybrid method for solving multi-objective global optimization problems. *Journal of Global Optimization*, **38**, 265–281. 8, 10

GIOSA, I., TANSINI, I., VIERA, I. *et al.* (2002). New assignment algorithms for the multi-depot vehicle routing problem. *Journal of the Operational Research Society*, **53**, 977–984. 30

GLOVER, F., KELLY, J. & LAGUNA, M. (1995). Genetic algorithms and tabu search: hybrids for optimization. *Computers & Operations Research*, **22**, 111–134. 16

GLOVER, F. *et al.* (1989). Tabu search – part I. *ORSA Journal on Computing*, **1**, 190–206. 16

GLYNN, P. & IGLEHART, D. (1989). Importance sampling for stochastic simulations. *Management Science*, 1367–1392. 19

GOLDBERG, D. (1989). *Genetic Algorithms in search, optimization and machine learning*. Addison-Wesley Publishing Company. 12, 25, 38

GONG, W., HO, Y. & ZHAI, W. (1992). Stochastic comparison algorithm for discrete optimization with estimation. In *Proceedings of the 31st IEEE Conference on Decision and Control, 1992*, 795–800, IEEE. 18

GREGOR, P., FORTHOFER, R. & KAPADIA, A. (1982). An evaluation of inventory and transportation policies of a regional blood distribution system. *European Journal of Operational Research*, **10**, 106–113. 60, 62, 67

HASTINGS, W. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57**, 97–109. 16

HAUMAN, C. & BEKKER, J. (2012). Application of the multi-objective cross-entropy method to the vehicle routing problem with soft time windows. *ORiON*, submitted. 28

HEMMELMAYR, V., DOERNER, K., HARTL, R. & SAVELSBERGH, M. (2009). Delivery strategies for blood products supplies. *OR spectrum*, **31**, 707–725. 63

HORN, J., NAFPLIOTIS, N. & GOLDBERG, D. (1994). A niched Pareto genetic algorithm for multi-objective optimization. In *IEEE World Congress on Computational Intelligence, Proceedings of the First IEEE Conference on Evolutionary Computation, 1994*, 82–87, IEEE. 16

IOANNOU, G., KRITIKOS, M. & PRASTACOS, G. (2003). A problem generator-solver heuristic for vehicle routing with soft time windows. *Omega*, **31**, 41–53. 31

JACOBS, C. (2004). *Operations Management for Competitive Advantage*. Boston: McGraw Hill. 64, 70

JAIMES, A., QUINTERO, L. & COELLO, C. (2009). Ranking methods in many-objective evolutionary algorithms. *Nature-Inspired Algorithms for Optimisation*, 413–434. 12

JENNINGS, J. (1973). Blood bank inventory control. *Management Science*, 637–645. 60, 61

JOINES, J. & ROBERTS, S. (2010). *Simulation modeling with SIMIO: a workbook*. SIMIO, LLC. 72

JONES, D., MIRRAZAVI, S. & TAMIZ, M. (2002). Multi-objective meta-heuristics: An overview of the current state-of-the-art. *European Journal of Operational Research*, **137**, 1–9. 14, 15

JOZEFOWIEZ, N., SEMET, F. & TALBI, E. (2008). Multi-objective vehicle routing problems. *European Journal of Operational Research*, **189**, 293–309. 33

KALLEHAUGE, B., LARSEN, J., MADSEN, O. & SOLOMON, M. (2005). Vehicle routing problem with time windows. *Column generation*, 67–98. 36, 37

KATSALIAKI, K. (2008). Cost-effective practices in the blood service sector. *Health policy*, **86**, 276–287. 59, 60, 62, 63

Katsaliaki, K. & Brailsford, S. (2006). Using simulation to improve the blood supply chain. *Journal of the Operational Research Society*, **58**, 219–227. 67

Kelton, W., Smith, J., Sturrock, D. & Verbraeck, A. (2010). *Simio and Simulation: Modeling, Analysis and Applications*. McGraw-Hill, Inc, New York, 1st edn. 66

Kendall, K. & Lee, S. (1980). Formulating blood rotation policies with multiple objectives. *Management Science*, 1145–1157. 60

Kirkpatrick, S., Gelatt Jr, C. & Vecchi, M. (1983). Optimization by simulated annealing. *Science*, **220**, 671–680. 16

Knowles, J. & Corne, D. (1999). The Pareto Archived Evolution Strategy: A new baseline algorithm for Pareto multi-objective optimisation. In *Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC 99*, vol. 1, IEEE. 16

Kuhn, H.W. & Tucker, A. (1951). Nonlinear programming. In J. Neyman, ed., *Proceedings of the Second Berkeley Simposium on Mathematical Statistics and Probability*, 481–492, University of California Press. 7

Law, A. & Kelton, W. (2000). *Simulation Modelling and Analysis*. Boston: McGraw-Hill. 72

Le Roux, J. (1981). *Die optimale ligging van bloedbanke*. Meesterstesis, Universiteit van Stellenbosch. 58

Lee, Y. & Park, B. (2006). Estimation of Kullback–Leibler divergence by local likelihood. *Annals of the Institute of Statistical Mathematics*, **58**, 327–340. 20

Ma, T. (2011). A cross-entropy multi-agent learning algorithm for solving vehicle routing problems with time windows. *Computational Logistics*, 59–73. 24, 35

Margolin, L. (2005). On the convergence of the cross-entropy method. *Annals of Operations Research*, **134**, 201–214. 20, 23

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, **21**, 1087. 16

Muller, A. (2008). Developing the idea of the thesis and the protocol. In L.O. Lategan, ed., *An Introduction to postgraduate supervision*, 41–66, AFRICAN SUN MeDIA. 4

OMBUKI, B., ROSS, B. & HANSHAR, F. (2006). Multi-objective genetic algorithms for vehicle routing problem with time windows. *Applied Intelligence*, **24**, 17–30. 33

PEGDEN, C. (2012). Intelligent objects: The future of simulation. http://www.simio.com/resources/white-papers/Intelligen-objects/Intelligent-Objects-The-Future-of-Simulation-Page-1.htm, accessed: 18 July 2012. 70, 72

PIDD, M. (1998). *Computer Simulation in Management Science*. Wiley, Chichester, UK, 4th edn. 67

PIERSKALLA, W. (2005). Supply chain management of blood banks. *Operations research and health care*, 103–145. 59

PRASTACOS, G. & BRODHEIM, E. (1979). Computer-based regional blood distribution. *Computers & Operations Research*, **6**, 69–77. 59, 60

RAAD, D., SINSKE, A. & VAN VUUREN, J. (2011). Water distribution systems design optimisation using metaheuristics and hyperheuristics. *ORiON*, **27**, 17–44. 43

RAUTONEN, J. (2007). Redesigning supply chain management together with the hospitals. *Transfusion*, **47**, 197S–200S. 63

ROBINSON, S. (2004). *Simulation: The Practice of Model Development and Use*. John Wiley & Sons Inc. 67, 68, 69

ROMEIJN, H. & SMITH, R. (1994). Simulated annealing for constrained global optimization. *Journal of Global Optimization*, **5**, 101–126. 16

RUBINSTEIN, R. (1997). Optimization of computer simulation models with rare events. *European Journal of Operational Research*, **99**, 89–112. 19

RUBINSTEIN, R. (1999). The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability*, **1**, 127–190, 10.1023/A:1010091220143. 19, 23

RUBINSTEIN, R.Y. & KROESE, D.P. (2004). *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*. Springer. 19, 21, 23, 24, 35

Rudolph, G. (2001). Evolutionary search under partially ordered fitness sets. In *Proceedings Of The International Symposium on Information Science Innovations in Engineering of Natural and Artificial Intelligent Systems (ISI 2001)*. 16

Rytilä, J. & Spens, K. (2006). Using simulation to increase efficiency in blood supply chains. *Management Research News*, **29**, 801–819. 61, 67

Sargent, R. (2005). Verification and validation of simulation models. In *Proceedings of the 37th Winter Simulation Conference*, 130–143. 69, 70

Sasaki, D. & Obayashi, S. (2005). Efficient search for trade-offs by adaptive range multi-objective genetic algorithms. *Journal of Aerospace Computing, Information, and Communication*, **2**, 44–64. 16

Savelsbergh, M. (1991). *The vehicle routing problem with time windows: Minimizing route duration*. Eindhoven University of Technology, Department of Mathematics and Computing Science. 31

Sawaragi, Y., Nakayama, H. & Tanino, T. (1985). *Theory of multi-objective optimization*, vol. 176. Elsevier Science. 8

Shi, L. & Ólafsson, S. (2000). Nested partitions method for global optimization. *Operations Research*, **48**, pp. 390–407. 18

Solomon, M.M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, **35**, pp. 254–265. 31, 32, 34

Steuer, R.E., Gardiner, L.R. & Gray, J. (1996). A bibliographic survey of the activities and international nature of multiple criteria decision making. *Journal of Multi-Criteria Decision Analysis*, **5**, 195–217. 8

Taillard, É., Badeau, P., Gendreau, M., Guertin, F. & Potvin, J. (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*, **31**, 170–186. 31, 32

Tan, K., Lee, L. & Ou, K. (2001). Artificial intelligence heuristics in solving vehicle routing problems with time window constraints. *Engineering Applications of Artificial Intelligence*, **14**, 825–837. 32

TAN, K., CHEW, Y. & LEE, L. (2006). A hybrid multi-objective evolutionary algorithm for solving vehicle routing problem with time windows. *Computational Optimization and Applications*, **34**, 115–151. 30, 33

TOTH, P. & VIGO, D. (2002). *The vehicle routing problem*, vol. 9. Society for Industrial Mathematics. 30

VILJOEN, N. (2011). Eighth annual state of logistics survey for South Africa 2011: Gearing up for change. 27

WHITE, K. & INGALLS, R. (2009). Introduction to simulation. In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, 12–23, IEEE. 69

YEGUL, M. (2007). *Simulation analysis of the blood supply chain and a case study*. Master's thesis, Middle East Technical University, Turkey. 61

ZITZLER, E. & THIELE, L. (1998). Multi-objective optimization using evolutionary algorithms: A comparative case study. In *Parallel problem solving from nature PPSN V*, 292–301, Springer. 16

ZITZLER, E., THIELE, L., LAUMANNS, M., FONSECA, C. & DA FONSECA, V. (2003). Performance assessment of multi-objective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, **7**, 117–132. 13, 43

APPENDIX A

RESULTS OF THE VRPSTW

This appendix contains the results of selected benchmark problems (Castro-Gutierrez *et al.*, 2011) of the vehicle routing problem with soft time windows as found by the cross-entropy algorithm for multi-objective optimisation. Two figures per problem and objective pairing are included, one to show the movement of the approximation front as the algorithm progressed and one to present the final approximation front. One result per problem is then documented in a table with routes to be travelled.

A.1 50_d0_tw1

V1	V2	V3	V4	V5	V6	V7	V8	V9
0	0	0	0	0	0	0	0	0
430761	430148	2138	430471	1897	2104	1813	1888	1389
1781	2152	430030	661	1777	1384	2003	430625	1870
2149	1721	1703	1509	1203	948	2107	1686	0
1235	856	0	1875	0	907	669	649	
2121	2000		430804		1714	974	0	
2044	430378		1725		2073	1362		
482	106		1588		486	0		
0	1173		0		2007			
	1463				1678			
	0				0			

Table A.1: Routes of solution A, 50_d0_tw1 ($Z1$ vs $Z3$).

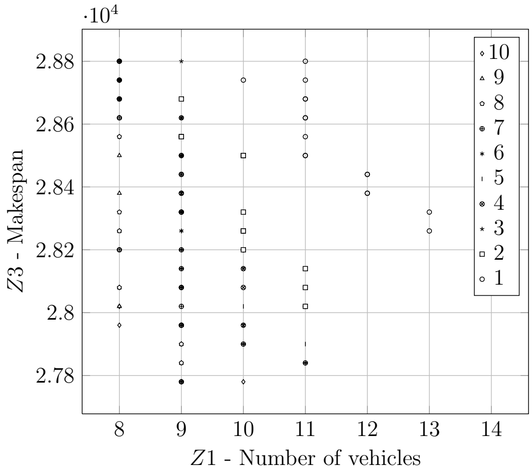

Figure A.1: Front progression of 50_d0_tw1 for $Z1$ vs $Z3$.

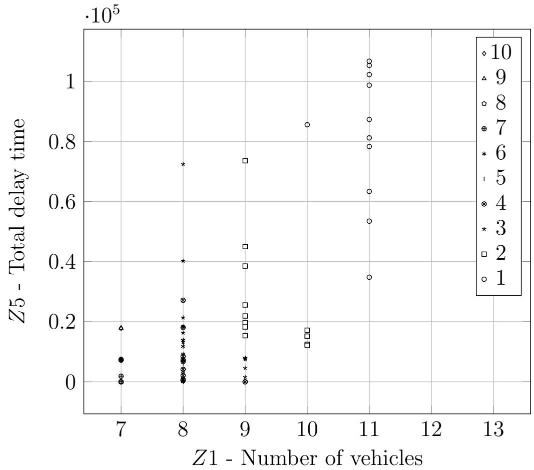

Figure A.2: Front progression of 50_d0_tw1 for $Z1$ vs $Z5$.

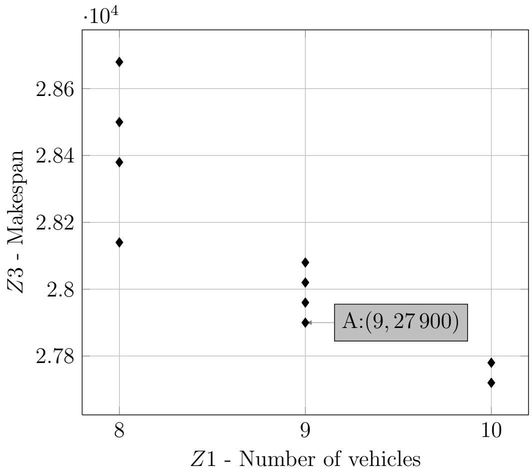

Figure A.3: Final approximation front of 50_d0_tw1 for $Z1$ vs $Z3$.

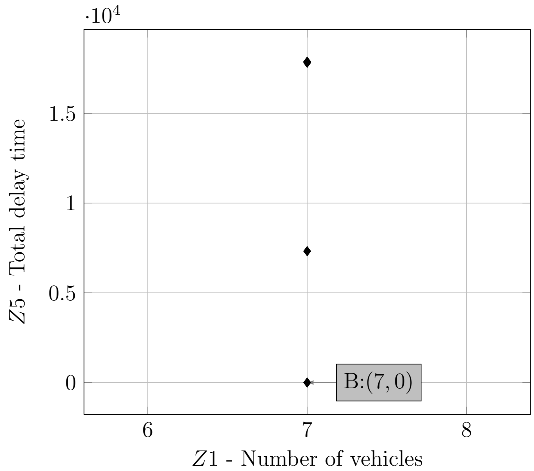

Figure A.4: Final approximation front of 50_d0_tw1 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7
0	0	0	0	0	0	0
1888	430761	1721	856	2000	430148	2138
106	1813	1897	661	1362	2152	2149
1678	2104	1384	1509	2107	1714	1703
1781	2003	430471	1777	1875	1173	2121
1389	1870	948	974	430625	1725	2044
1235	669	907	482	1686	2073	0
1203	430804	486	0	649	0	
430030	0	2007		430378		
0		1588		0		
		1463				
		0				

Table A.2: Routes of solution B, 50_d0_tw1 ($Z1$ vs $Z5$).

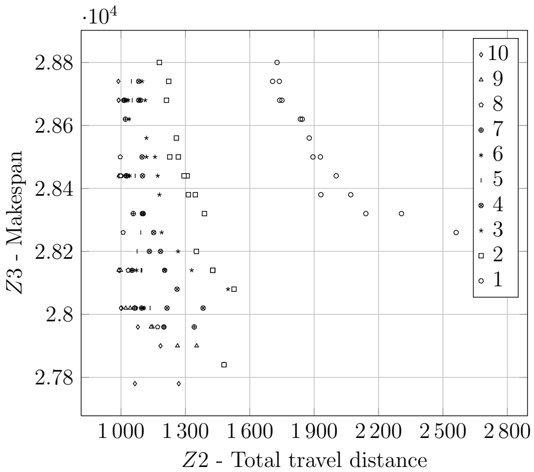

Figure A.5: Front progression of 50_d0_tw1 for $Z2$ vs $Z3$.

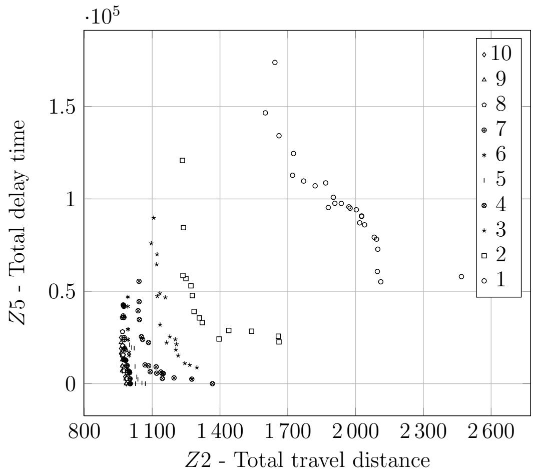

Figure A.6: Front progression of 50_d0_tw1 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
2003	1389	430030	1875	1813	430471	1714	430761
482	1870	1703	2149	2104	2000	948	2107
1725	2121	0	1235	661	1362	1384	2138
1588	2044		1781	856	1721	2073	1203
0	0	0	669	907	2152	1897	1777
			430804	1509	1888	430625	0
			974	1678	430148	1173	
			0	2007	1686	0	
				486	649		
				430378	1463		
				106	0		
				0			

Table A.3: Routes of solution C, 50_d0_tw1 ($Z2$ vs $Z3$).

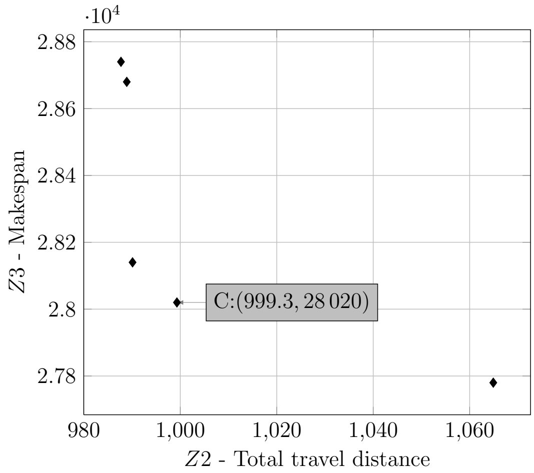

Figure A.7: Final approximation front of 50_d0_tw1 for $Z2$ vs $Z3$.

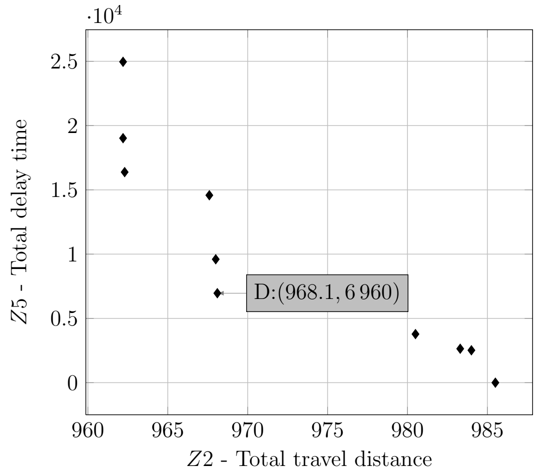

Figure A.8: Final approximation front of 50_d0_tw1 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
1875	430471	430761	1813	106	1362	2000	1714
1235	856	2107	2104	948	430148	1888	430378
2149	661	669	2003	1897	2138	1703	0
1781	907	430804	482	430625	1203	430030	
1389	1509	974	1725	1721	1777	0	
2121	1678	0	1588	2152	0		
2044	1384		0	1686			
1870	1173			649			
0	2007			2073			
	486			0			
	1463						
	0						

Table A.4: Routes of solution D, 50_d0_tw1 ($Z2$ vs $Z5$).

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
430761	2107	1888	2000	1235	1678	430148	1389
2152	1509	2149	2003	907	2138	1875	1721
430625	1781	661	1897	1714	430471	1362	1777
1384	856	1203	1813	2073	2044	948	1725
2104	649	1870	669	430378	2121	430030	0
1686	1173	0	430804	1588	482	1703	
486	2007		974	0	0	0	
1463	106		0				
0	0						

Table A.5: Routes of solution E, 50_d0_tw1 ($Z4$ vs $Z3$).

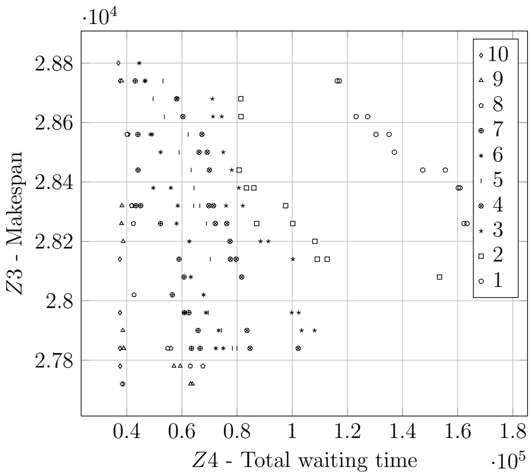

Figure A.9: Front progression of 50_d0_tw1 for $Z4$ vs $Z3$.

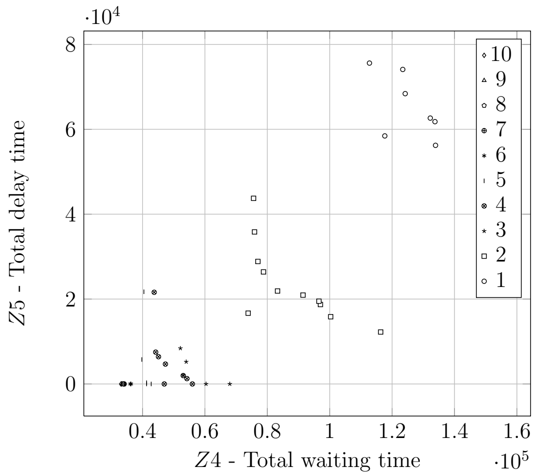

Figure A.10: Front progression of 50_d0_tw1 for $Z4$ vs $Z5$.

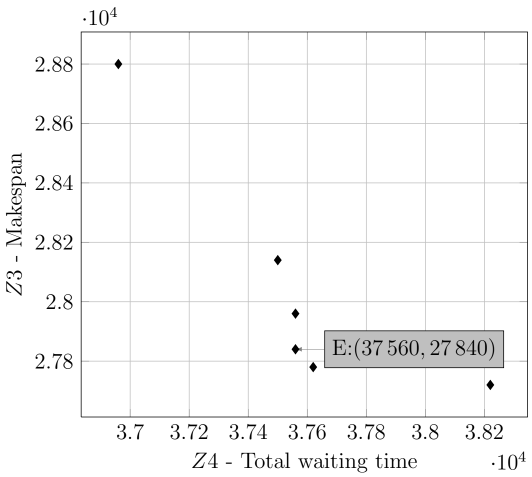

Figure A.11: Final approximation front of 50_d0_tw1 for $Z4$ vs $Z3$.

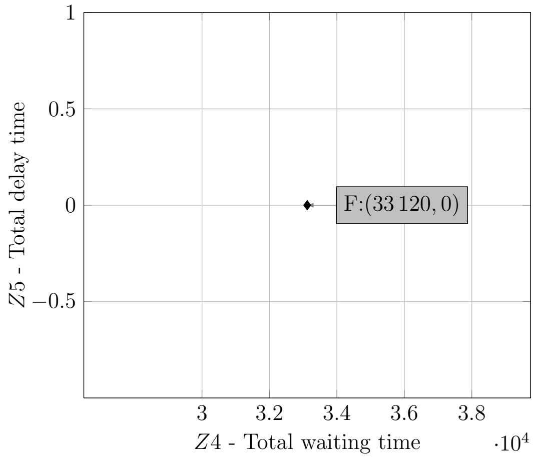

Figure A.12: Final approximation front of 50_d0_tw1 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
1888	1362	2107	1389	430148	430761	106	1678
2104	2138	430471	1714	1235	2000	1875	2149
948	1897	2152	907	1509	1781	1721	1813
1384	669	661	1686	856	430625	1203	430030
2003	430804	1777	1173	2121	1703	649	1870
430378	1588	2044	1463	482	2007	0	0
2073	0	974	0	486	0		
1725		0		0			
0							

Table A.6: Routes of solution F, 50_d0_tw1 ($Z4$ vs $Z5$).

A.2 50_d0_tw2

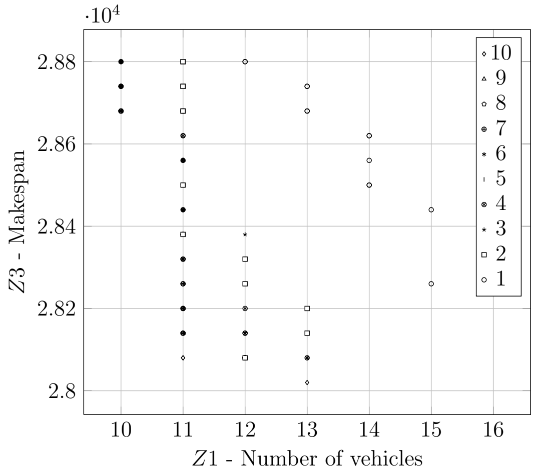

Figure A.13: Front progression of 50_d0_tw2 for $Z1$ vs $Z3$.

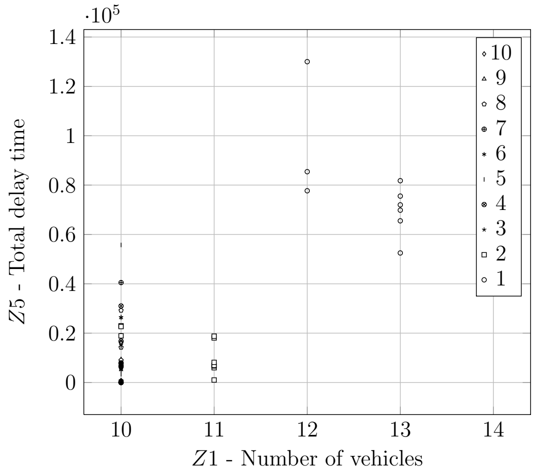

Figure A.14: Front progression of 50_d0_tw2 for $Z1$ vs $Z5$.

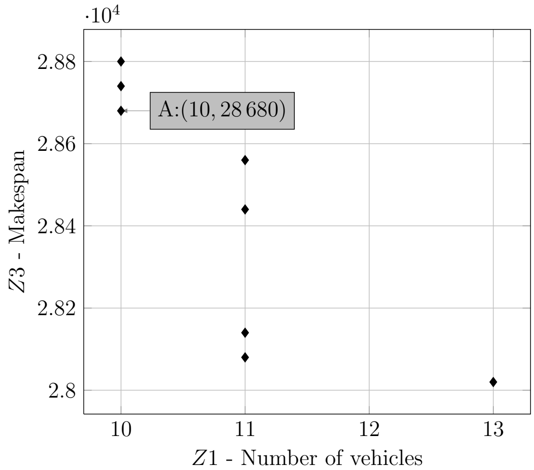

Figure A.15: Final approximation front of 50_d0_tw2 for $Z1$ vs $Z3$.

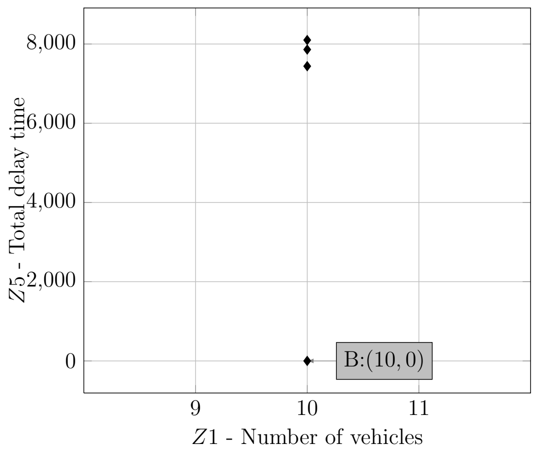

Figure A.16: Final approximation front of 50_d0_tw2 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
2138	2003	1813	430148	1509	2104	2152	661
1203	1897	1714	1389	1686	2149	1721	2000
974	1362	1173	1703	1725	1870	1888	1781
0	430471	1463	482	0	2121	856	2044
	948	1588	0	0		907	1777
	1384	0				669	0
	649					2107	
	430378					430804	
	106					0	
	0						

Table A.7: Routes of solution A, 50_d0_tw2 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
0	0	0	0	0	0	0	0	0	0
2000	1384	1678	1362	430625	1888	430148	1173	2107	430471
1721	486	1870	106	1897	907	1714	2073	430761	2152
2138	2007	430804	2003	1686	2104	2149	1588	1235	948
1703	430378	0	1781	1725	1813	1389	0	1875	856
1777	0		669	0	1509	1203	0	2121	649
0			2044		661	974		482	1463
			0		430030	0		0	0
					0				

Table A.8: Routes of solution B, 50_d0_tw2 ($Z1$ vs $Z5$).

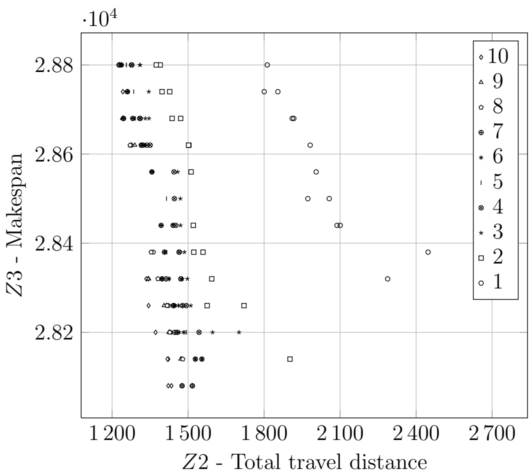

Figure A.17: Front progression of 50_d0_tw2 for $Z2$ vs $Z3$.

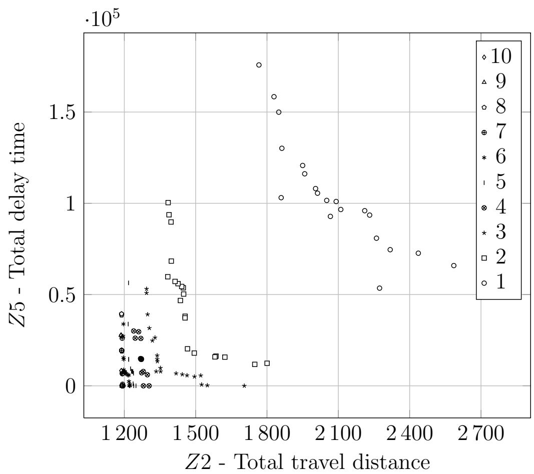

Figure A.18: Front progression of 50_d0_tw2 for $Z2$ vs $Z5$.

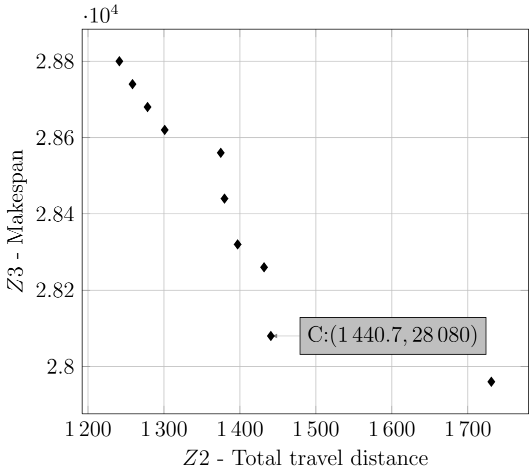

Figure A.19: Final approximation front of 50_d0_tw2 for $Z2$ vs $Z3$.

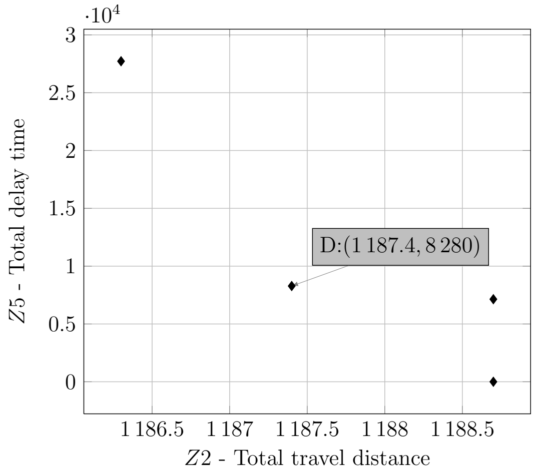

Figure A.20: Final approximation front of 50_d0_tw2 for $Z2$ vs $Z5$.

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**
0	0	0	0	0	0	0	0	0	0	0
1725	430471	1897	2003	1509	2138	1875	1813	1870	430030	482
1588	2073	1721	1389	948	1203	1235	2104	2121	0	974
1714	486	1362	1703	1384	430761	1781	856	0		0
0	1463	2000	2149	430625	0	669	661			
	0	649	0	2152	0	2107	907	0		
		430378		1686		430804	1173			
		106		430148		0	2007			
		0		1888			0			
				1678						
				0						

Table A.9: Routes of solution C, 50_d0_tw2 ($Z2$ vs $Z3$).

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**
0	0	0	0	0	0	0	0	0
2003	2104	2107	106	1781	1678	2000	1725	1235
1870	1813	430761	1362	2138	948	1721	1588	1875
482	856	669	1888	1203	1897	2152	0	2149
0	661	430804	430148	974	430625	1686		1389
	907	0	1703	0	430030	649		2121
	1509		1777		430471	0		2044
	2007		0		0			
	486							
	430378							
	0							

Table A.10: Routes of solution D, 50_d0_tw2 ($Z2$ vs $Z5$).

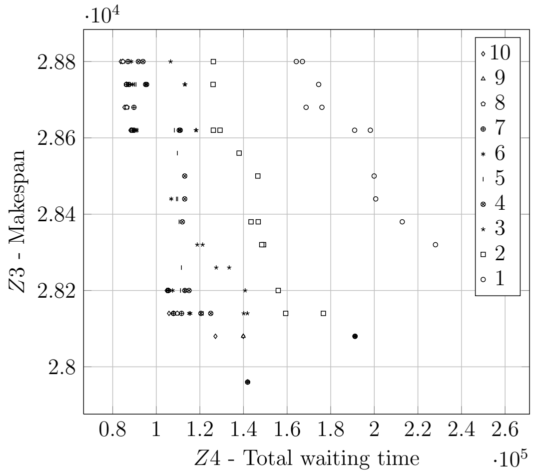

Figure A.21: Front progression of 50_d0_tw2 for $Z4$ vs $Z3$.

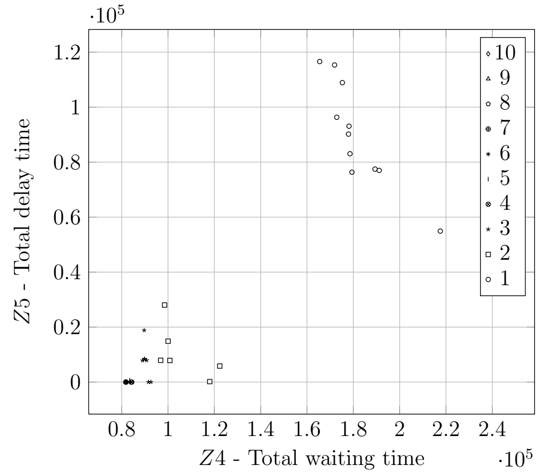

Figure A.22: Front progression of 50_d0_tw2 for $Z4$ vs $Z5$.

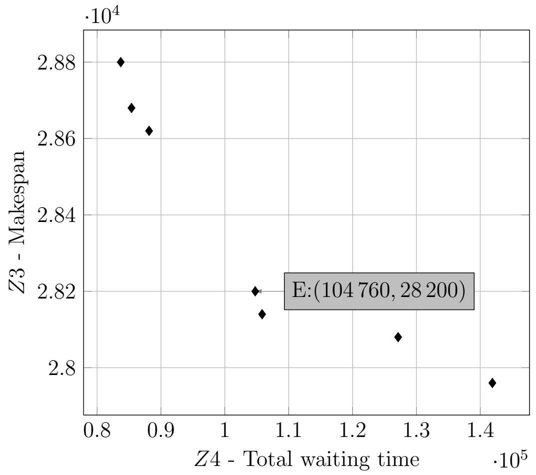

Figure A.23: Final approximation front of 50_d0_tw2 for $Z4$ vs $Z3$.

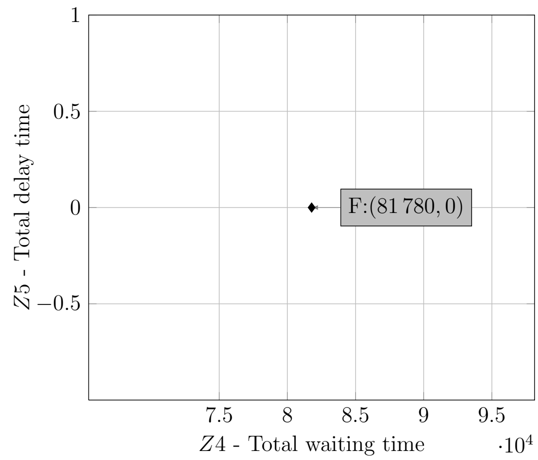

Figure A.24: Final approximation front of 50_d0_tw2 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11
0	0	0	0	0	0	0	0	0	0	0
2107	430148	1389	1362	1781	2138	661	1888	2000	1703	1384
2104	430761	2152	2149	856	430471	948	1875	1235	0	1203
1509	907	669	1897	2003	1777	1870	1721	1686		0
649	1714	974	430030	430378	430804	2121	1813	1588		
1173	1678	0	106	1725	0	0	2073	0		
1463	430625		0	0			2007			
0	2044						486			
	482						0			
	0									

Table A.11: Routes of solution E, 50_d0_tw2 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
0	0	0	0	0	0	0	0	0	0
1721	2000	1235	1362	1888	430148	2138	1678	1897	948
1875	106	907	1781	430761	1389	661	2107	1714	1813
649	2003	1384	856	430471	1509	1203	430625	2104	430030
2073	2152	1777	430804	669	1870	974	2149	430378	0
486	1703	1463	482	1725	2121	0	1686	2007	
0	2044	0	0	0	0	0	1588	1173	
	0						0	0	

Table A.12: Routes of solution F, 50_d0_tw2 ($Z4$ vs $Z5$).

A.3 50_d0_tw3

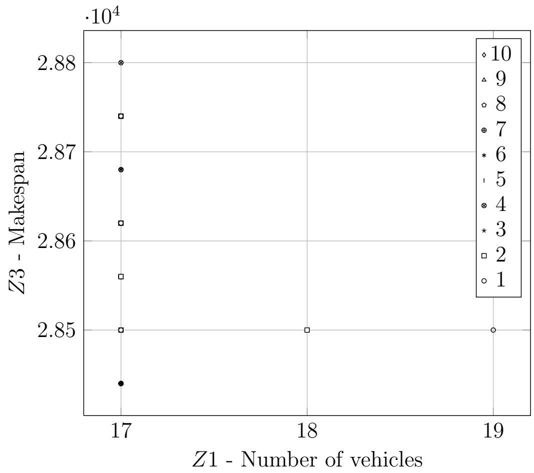

Figure A.25: Front progression of 50_d0_tw3 for $Z1$ vs $Z3$.

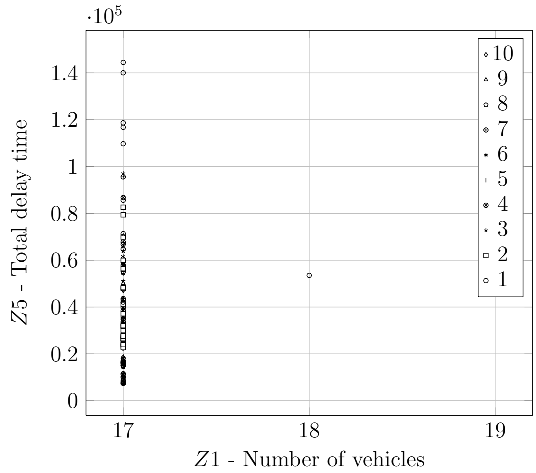

Figure A.26: Front progression of 50_d0_tw3 for $Z1$ vs $Z5$.

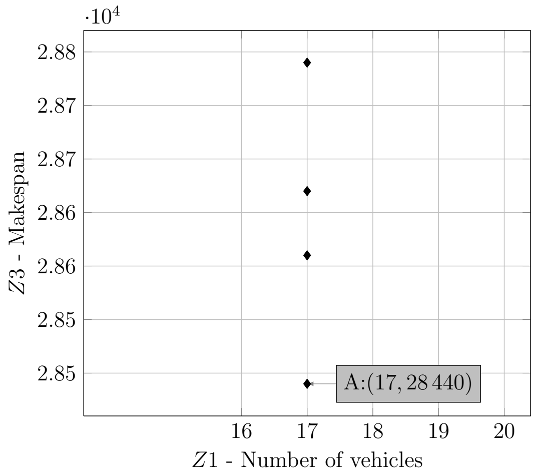

Figure A.27: Final approximation front of 50_d0_tw3 for $Z1$ vs $Z3$.

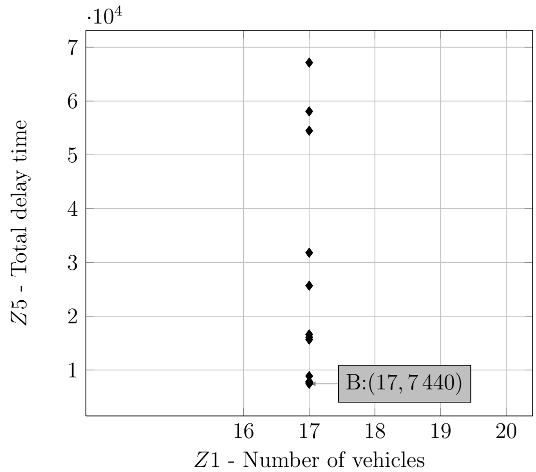

Figure A.28: Final approximation front of 50_d0_tw3 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1714	1888	1509	430148	1813	2149	974	1777	2000	1721	2003	430804	2104	430625	1463	2138	430030
907	661	1389	106	948	1875	430761	0	1686	1362	669	0	1870	2044	2152	856	0
430471	1725	1703	1781	1384	1203	0		1897	2007	2107		0	0	0	2121	
482	1173	0	1235	649	0				1588	0					0	
1678	0		430378	486					0							
0			2073	0												
			0													

Table A.13: Routes of solution A, 50_d0_tw3 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1678	430148	1362	430804	2000	2044	106	649	2152	2104	1813	2149	1888	1384	2073	2138	974
430761	1897	1725	1781	661	0	430625	1173	1203	669	1509	1703	948	856	0	1686	0
2107	1714	1463	0	1721		1588	0	0	0	1777	0	430471	907		0	
1235	486	0		430030		2007				0		1870	2003			
1875	430378			0		0						0	482			
1389	0												0			
2121																
0																

Table A.14: Routes of solution B, 50_d0_tw3 ($Z1$ vs $Z5$).

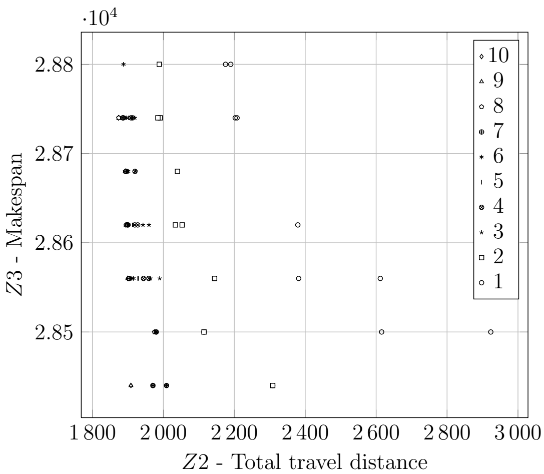

Figure A.29: Front progression of 50_d0_tw3 for $Z2$ vs $Z3$.

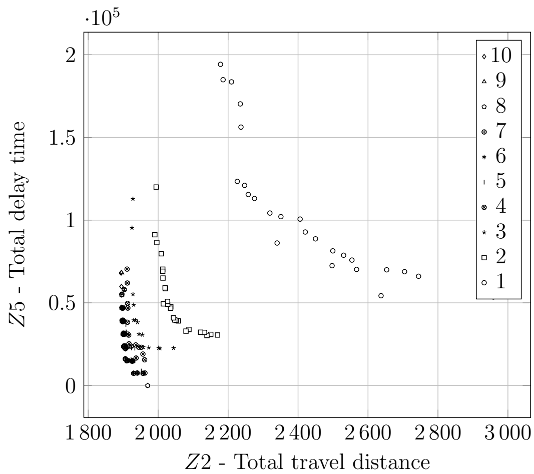

Figure A.30: Front progression of 50_d0_tw3 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1781	2138	1714	1725	430471	2107	1777	974	948	1870	2044	106	2152	1389	482	2104	430030
430804	1203	430378	1588	1721	669	0	2149	1384	0	1235	661	1686	2121	2003	1813	0
1875	0	1173	0	430148	430761		0	1897		0	856	1888	0	0	1463	
0		0		1703	0			430625			907	1362			0	
				0				2000			1509	0				
								649			1678					
								2073			2007					
								0			486					
											0					

Table A.15: Routes of solution C, 50_d0_tw3 ($Z2$ vs $Z3$).

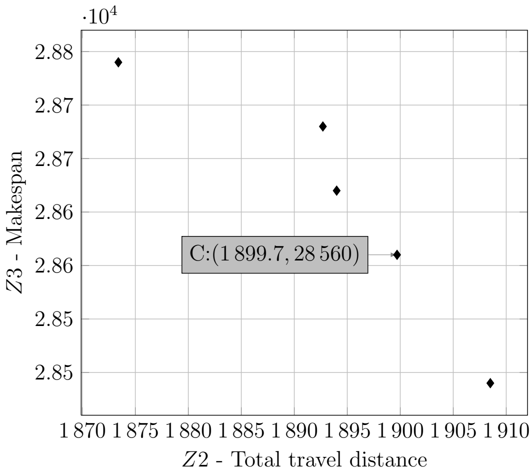

Figure A.31: Final approximation front of 50_d0_tw3 for $Z2$ vs $Z3$.

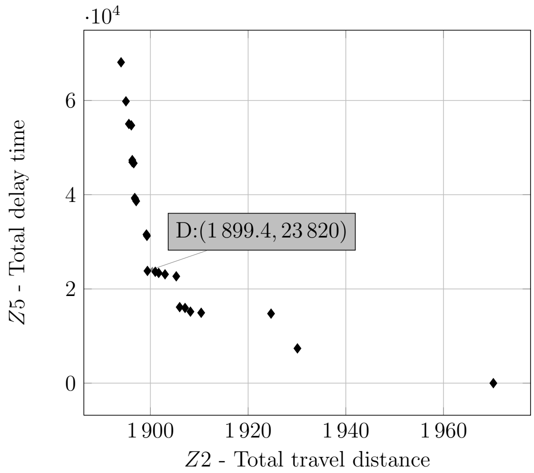

Figure A.32: Final approximation front of 50_d0_tw3 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1813	1389	1870	106	2107	1777	2044	1714	2000	1362	1725	2149	1897	430471	430804	2003	1463
2104	2121	0	1678	430761	0	1235	430030	1888	430148	1588	974	430625	948	0	482	0
1384	1875		661	1781		0	0	2138	1703	0	0	2152	1509		0	
2073	0		907	669				1203	0			1686	2007			
649			856	0				0				1721	1173			
0			486									0	0			
			430378													
			0													

Table A.16: Routes of solution D, 50_d0_tw3 ($Z2$ vs $Z5$).

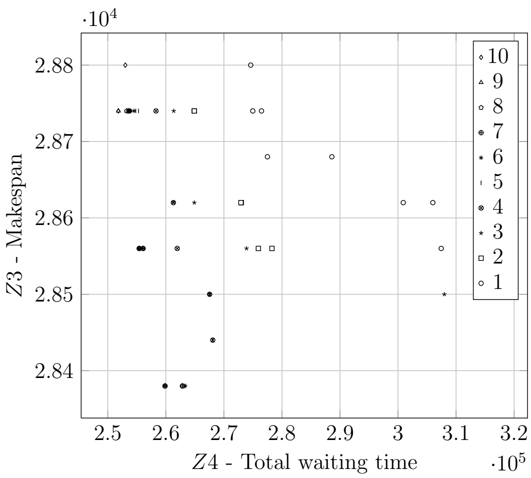

Figure A.33: Front progression of 50_d0_tw3 for $Z4$ vs $Z3$.

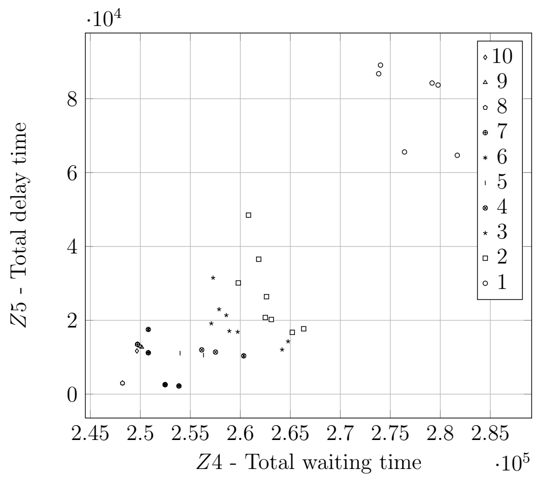

Figure A.34: Front progression of 50_d0_tw3 for $Z4$ vs $Z5$.

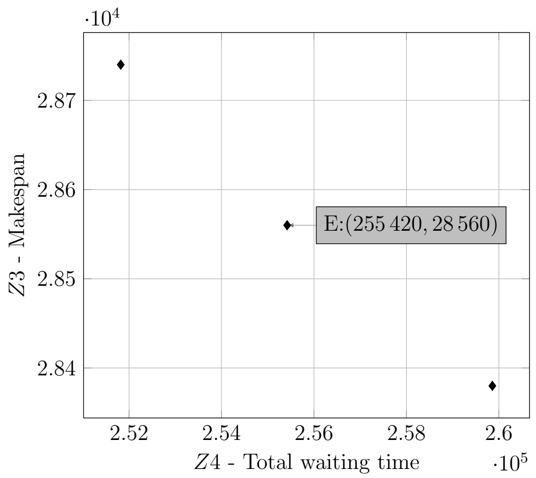

Figure A.35: Final approximation front of 50_d0_tw3 for $Z4$ vs $Z3$.

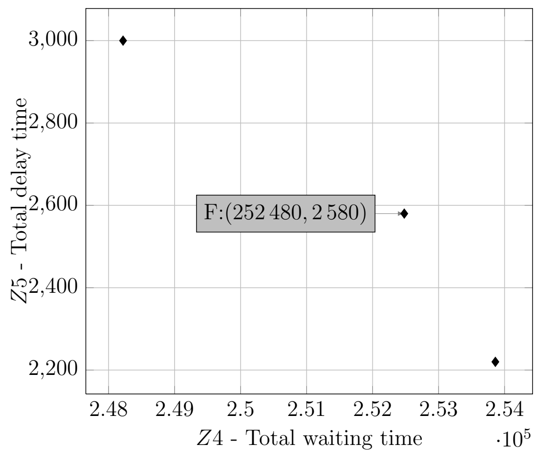

Figure A.36: Final approximation front of 50_d0_tw3 for $Z4$ vs $Z5$.

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**	**V13**	**V14**	**V15**	**V16**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430761	1235	1721	2138	2149	1714	2107	430471	1389	430804	1703	1509	1888	1362	1813	1203
661	430148	2003	430625	1686	1781	2152	907	856	0	0	1897	669	482	1463	0
1875	974	2000	430378	948	2104	1725	1870	1777			1384	0	1678	0	
649	106	1588	2007	0	2044	1173	0	0			2121		0		
486	0	2073	0		0	0					0				
0		0													

Table A.17: Routes of solution E, 50_d0_tw3 ($Z4$ vs $Z3$).

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**	**V13**	**V14**	**V15**	**V16**	**V17**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1888	1678	2107	430471	106	2138	907	1362	430148	2044	2003	2152	1686	486	430030	974	669
430761	1897	2000	1703	1235	649	1203	1714	1389	0	1384	1509	0	0	0	0	0
430625	1721	1875	0	2073	1173	0	2104	661		1870	482					
2149	1725	1781		1588	0		430378	948		0	0					
856	2007	1813		0			1463	430804								
2121	0	1777					0	0								
0		0														

Table A.18: Routes of solution F, 50_d0_tw3 ($Z4$ vs $Z5$).

A.4 50_d0_tw4

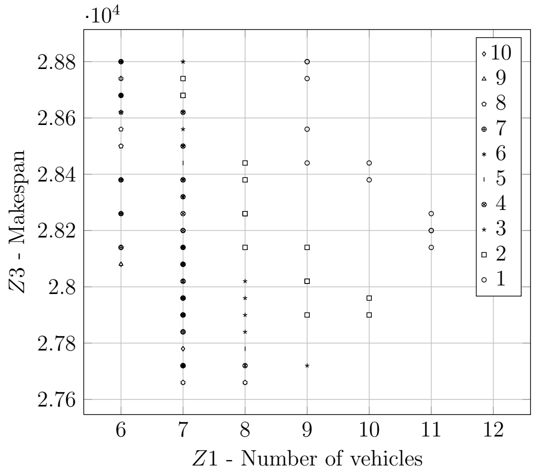

Figure A.37: Front progression of 50_d0_tw4 for $Z1$ vs $Z3$.

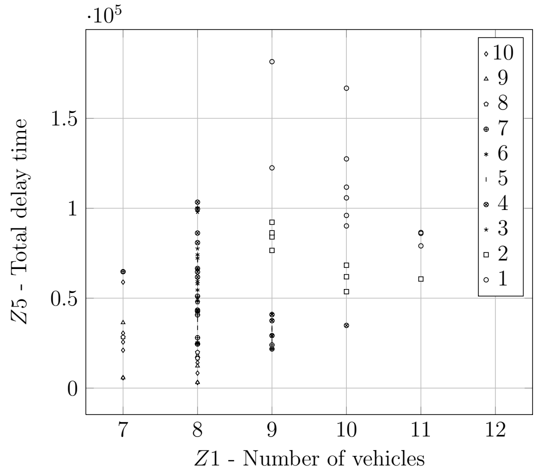

Figure A.38: Front progression of 50_d0_tw4 for $Z1$ vs $Z5$.

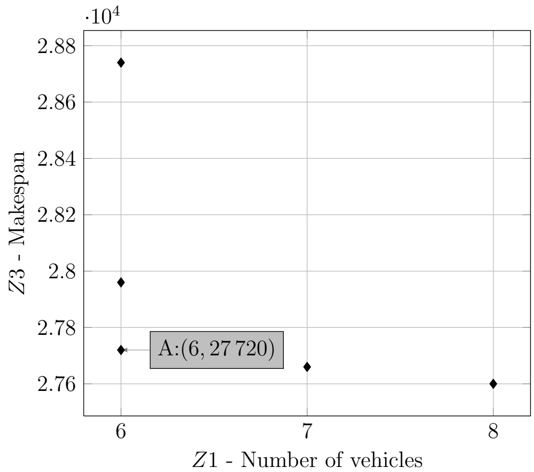

Figure A.39: Final approximation front of 50_d0_tw4 for $Z1$ vs $Z3$.

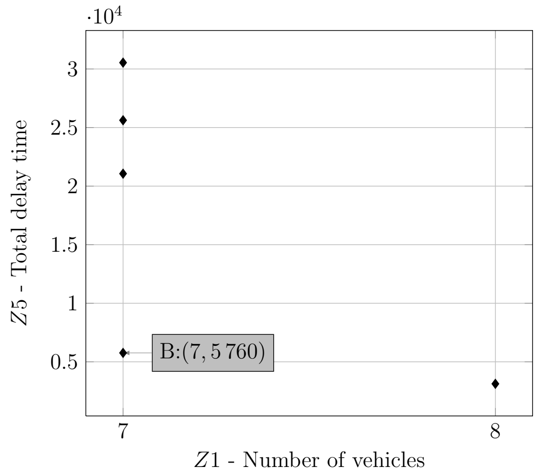

Figure A.40: Final approximation front of 50_d0_tw4 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
430148	1870	1777	1714	649	430378
1725	1203	486	1173	430030	1588
430761	1389	1463	2000	1703	907
2107	2121	1721	2073	2044	1897
1875	974	1362	856	2149	661
1235	1781	1384	1509	669	1686
430804	2138	2007	1678	2003	1888
482	0	2104	106	0	1813
0		0	430471		0
			948		
			430625		
			2152		
			0		

Table A.19: Routes of solution A, 50_d0_tw4 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7
0	0	0	0	0	0	0
1870	1888	1777	1897	856	649	1725
430804	1714	1389	2104	1173	1813	2107
1678	106	2044	1384	430378	1509	430030
948	1588	430761	430148	2121	2138	1875
0	1463	1703	661	2149	0	974
	2073	669	1203	2003		486
	907	482	1235	430471		2007
	1686	0	1781	0		2152
	2000		1721			0
	430625		1362			
	0		0			

Table A.20: Routes of solution B, 50_d0_tw4 ($Z1$ vs $Z5$).

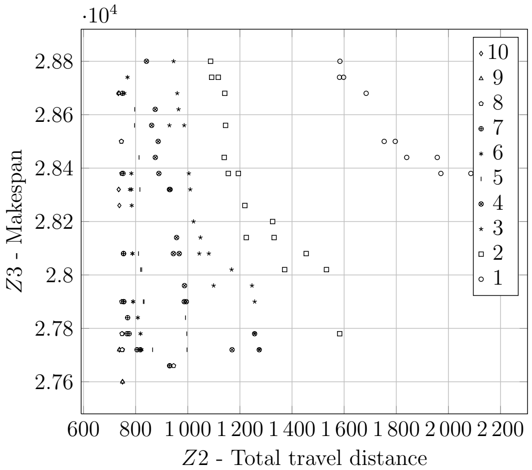

Figure A.41: Front progression of 50_d0_tw4 for $Z2$ vs $Z3$.

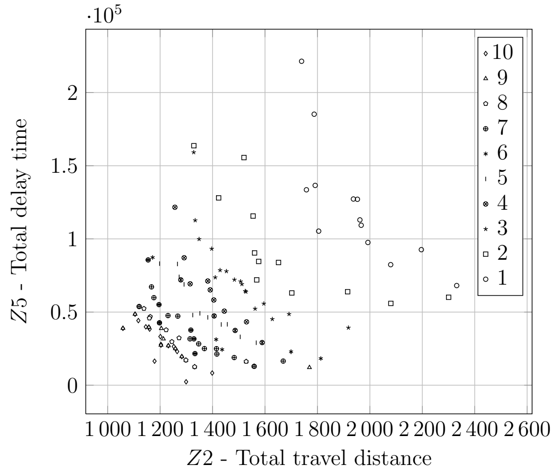

Figure A.42: Front progression of 50_d0_tw4 for $Z2$ vs $Z5$.

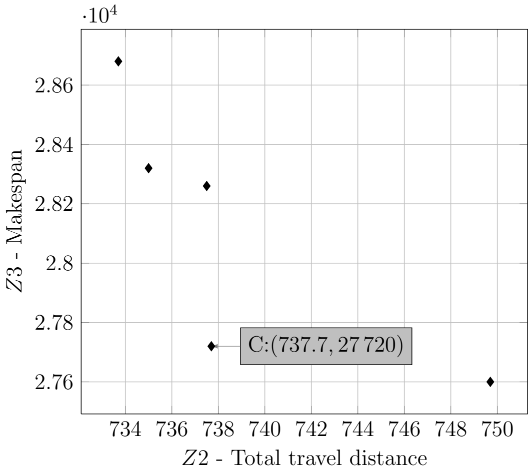

Figure A.43: Final approximation front of 50_d0_tw4 for $Z2$ vs $Z3$.

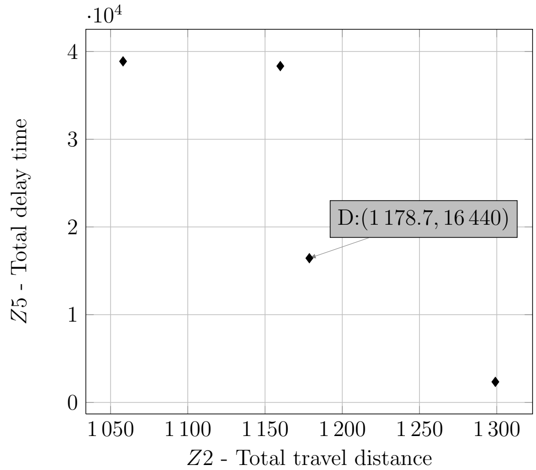

Figure A.44: Final approximation front of 50_d0_tw4 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
1725	430471	1588	1870	1714	1813
1235	2073	1463	2121	430378	2104
1875	1897	486	2044	430030	1173
974	430625	856	1389	1703	1384
430761	106	661	2149	1777	649
2107	0	907	669	1203	1362
1781		1509	2003	2138	2000
430804		948	0	0	430148
482		1678			1686
0		2007			1888
		0			2152
					1721
					0

Table A.21: Routes of solution C, 50_d0_tw4 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
1870	430625	2104	856	486	1897	661	1678
1777	2007	1813	430378	2152	1725	2073	1509
2107	948	1888	1714	1362	1173	907	0
1235	0	1875	1384	2000	1686	1463	
2044		1389	649	1721	430148	1588	
2121		430761	430030	0	430471	1781	
974		2149	1703		0	0	
430804		669	1203				
2003		482	2138				
106		0	0				
0							

Table A.22: Routes of solution D, 50_d0_tw4 ($Z2$ vs $Z5$).

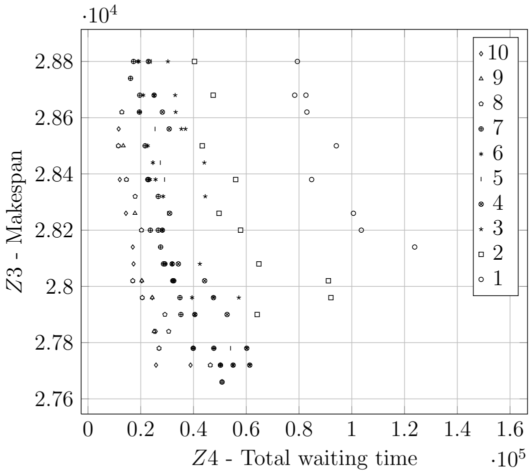

Figure A.45: Front progression of 50_d0_tw4 for $Z4$ vs $Z3$.

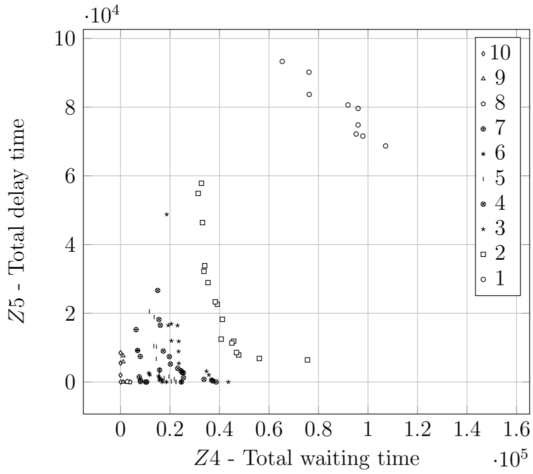

Figure A.46: Front progression of 50_d0_tw4 for $Z4$ vs $Z5$.

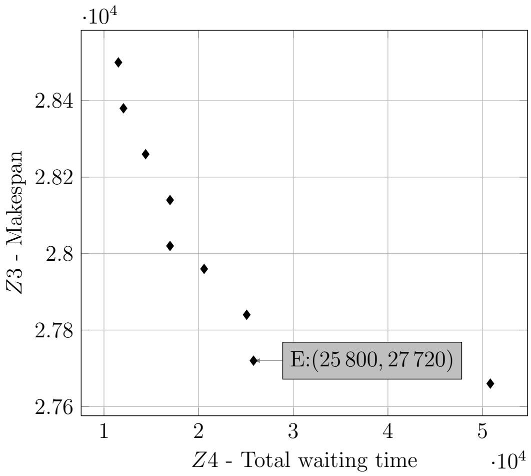

Figure A.47: Final approximation front of 50_d0_tw4 for $Z4$ vs $Z3$.

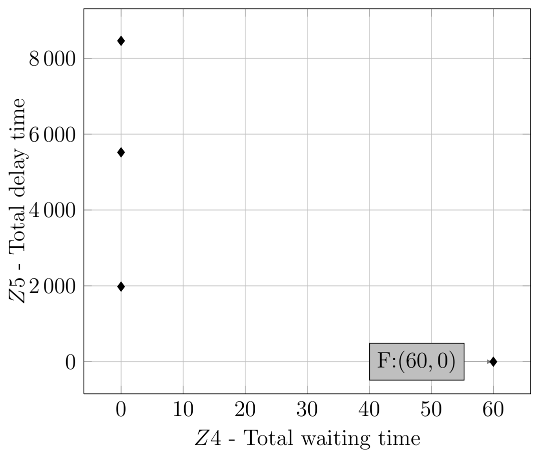

Figure A.48: Final approximation front of 50_d0_tw4 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7
0	0	0	0	0	0	0
430148	430030	2107	1870	1777	430378	1384
649	1875	2104	1173	856	2073	1725
2121	1389	1703	2044	1588	1897	1888
1463	1781	430761	1235	2149	430804	1714
661	669	486	1721	2138	482	1203
1678	2003	1362	1686	0	0	974
106	0	2152	430625			430471
907		1813	0			948
2007		0				2000
1509						0
0						

Table A.23: Routes of solution E, 50_d0_tw4 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7
0	0	0	0	0	0	0
430378	2000	1897	1870	1173	856	1725
1384	1813	1714	1362	1777	2104	0
2107	649	430148	1875	661	1888	
430030	1203	974	1235	907	1703	
1509	486	2073	2044	1588	430761	
1463	669	106	2121	2149	1389	
2138	2003	1781	948	482	1686	
0	0	430804	1678	430471	2152	
		0	2007	0	1721	
			430625		0	
			0			

Table A.24: Routes of solution F, 50_d0_tw4 ($Z4$ vs $Z5$).

A.5 50_d1_tw1

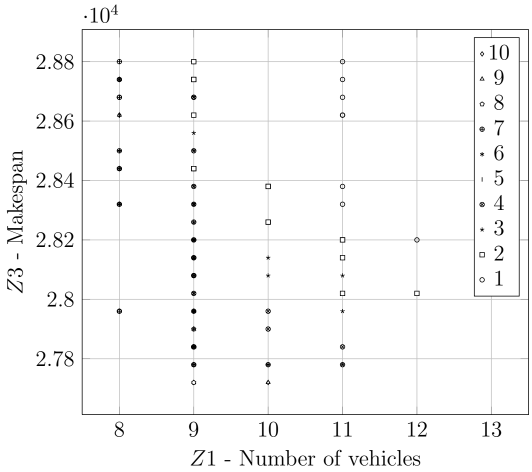

Figure A.49: Front progression of 50_d1_tw1 for $Z1$ vs $Z3$.

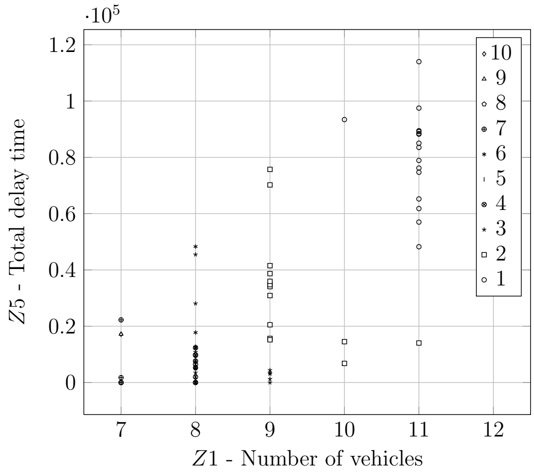

Figure A.50: Front progression of 50_d1_tw1 for $Z1$ vs $Z5$.

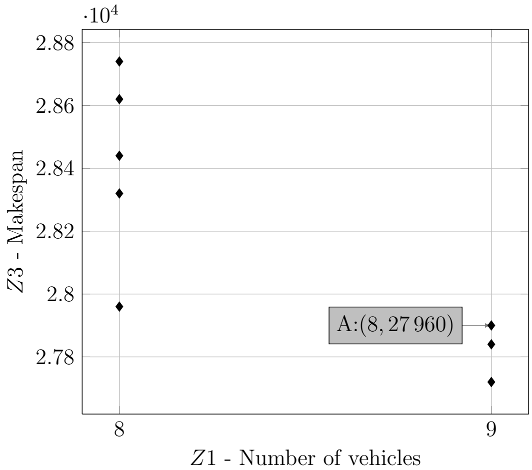

Figure A.51: Final approximation front of 50_d1_tw1 for $Z1$ vs $Z3$.

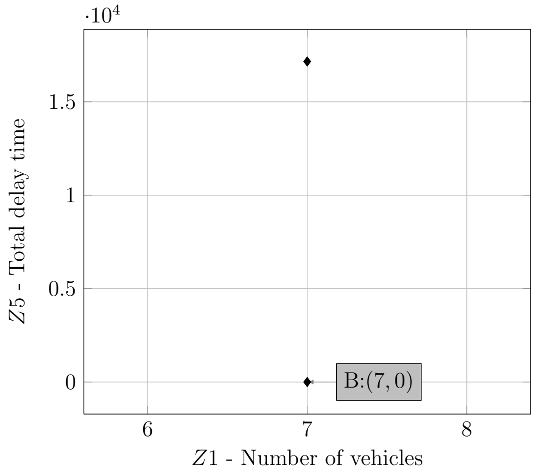

Figure A.52: Final approximation front of 50_d1_tw1 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
430761	2138	1888	1721	1714	2104	2003	2149
2107	430030	430148	1362	430471	907	2152	2121
1235	1703	1875	1678	430625	661	1203	2044
1813	0	1781	856	1509	1897	486	1463
948		1389	2000	1384	669	0	0
1686		1870	430804	106	974		
649		1777	1725	430378	482		
1173		0	1588	2073	0		
0			0	2007			
				0			

Table A.25: Routes of solution A, 50_d1_tw1 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7
0	0	0	0	0	0	0
106	1362	907	2104	430625	2000	2107
1714	1813	661	1384	948	1888	430761
430471	1777	1721	1509	1725	430148	2138
2003	430804	2152	856	1588	1678	1389
430030	482	1686	1897	430378	1781	1870
1203	0	1463	2073	0	1875	669
0		649	486		1235	974
		0	2007		2149	0
			1173		1703	
			0		2121	
					2044	
					0	

Table A.26: Routes of solution B, 50_d1_tw1 ($Z1$ vs $Z5$).

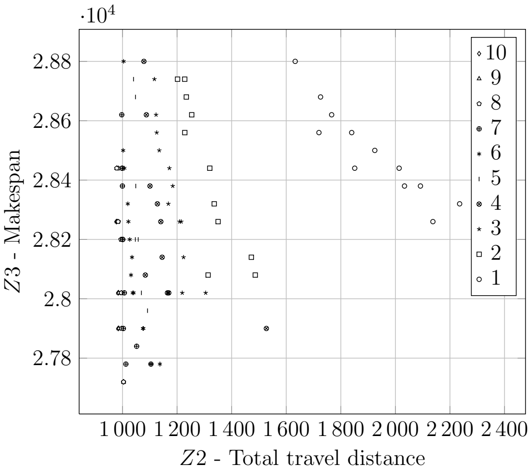

Figure A.53: Front progression of 50_d1_tw1 for $Z2$ vs $Z3$.

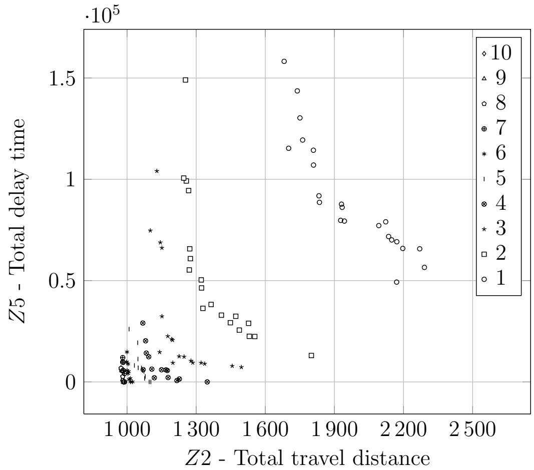

Figure A.54: Front progression of 50_d1_tw1 for $Z2$ vs $Z5$.

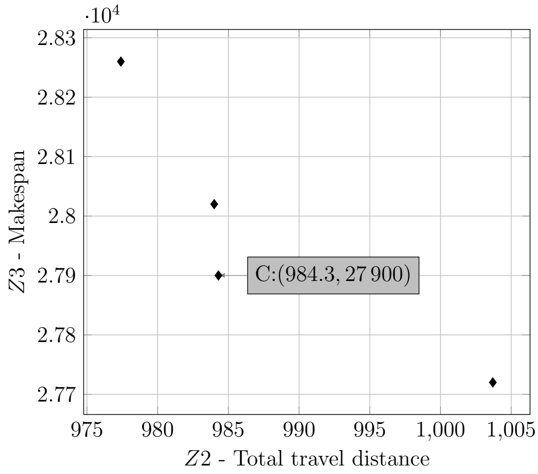

Figure A.55: Final approximation front of 50_d1_tw1 for $Z2$ vs $Z3$.

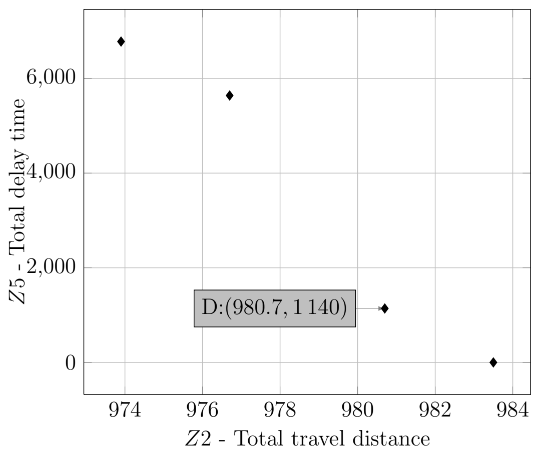

Figure A.56: Final approximation front of 50_d1_tw1 for $Z2$ vs $Z5$.

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**
0	0	0	0	0	0	0	0
1714	2152	1813	430030	482	1781	1875	2003
106	1721	2104	1703	1725	2107	1235	1870
430471	1362	1384	0	1588	430761	2149	2121
661	1686	1897		0	669	1389	2044
856	430148	430625			430804	2138	0
907	1888	2073			974	1203	
1509	649	1173			0	1777	
1678	2000	948					
486	0	1463					
2007		0					
430378							
0							

Table A.27: Routes of solution C, 50_d1_tw1 ($Z2$ vs $Z3$).

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**
0	0	0	0	0	0	0	0
1897	1235	1813	106	2107	2000	2003	1714
430625	1875	2104	1678	430761	430148	482	430471
1721	2149	907	1362	1781	2138	1725	430378
2152	1389	856	1888	430804	1203	1588	1463
1686	1870	661	1703	669	1777	0	0
649	2121	1509	430030	974	0		
2073	2044	948	0	0			
0	0	1384					
		1173					
		2007					
		486					
		0					

Table A.28: Routes of solution D, 50_d1_tw1 ($Z2$ vs $Z5$).

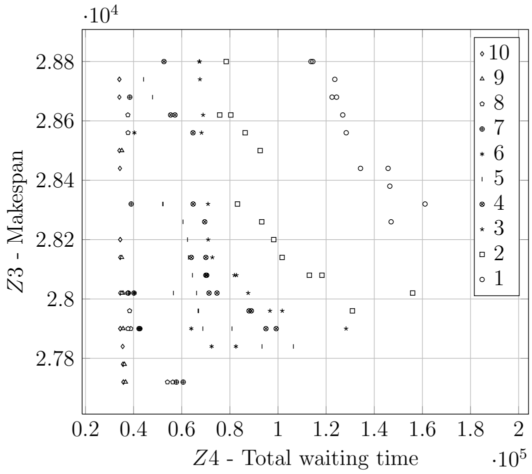

Figure A.57: Front progression of 50_d1_tw1 for $Z4$ vs $Z3$.

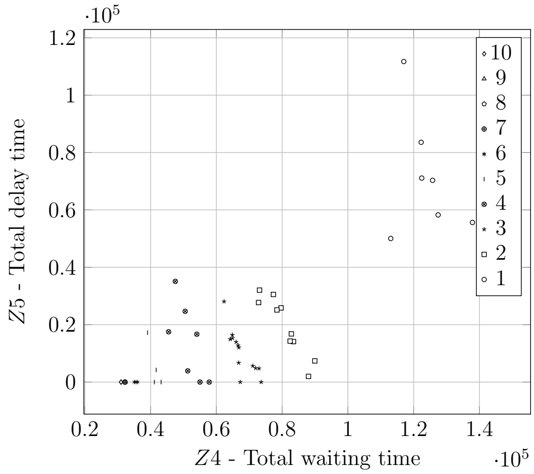

Figure A.58: Front progression of 50_d1_tw1 for $Z4$ vs $Z5$.

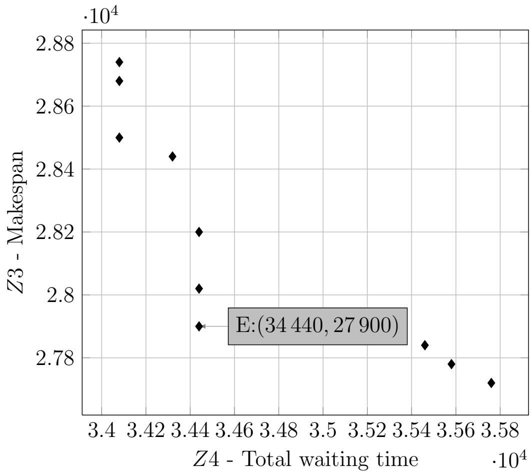

Figure A.59: Final approximation front of 50_d1_tw1 for $Z4$ vs $Z3$.

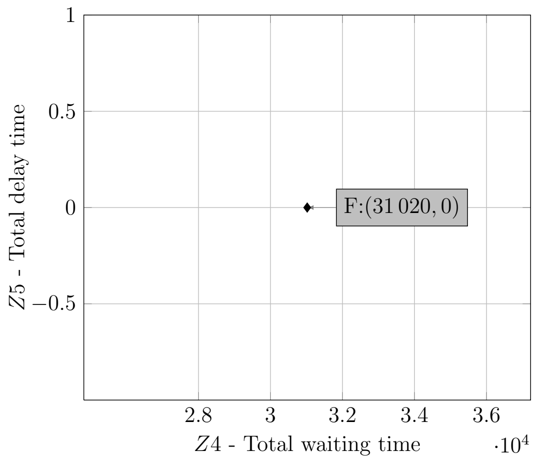

Figure A.60: Final approximation front of 50_d1_tw1 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
1678	661	430148	1362	1235	430761	2107	1888
1781	1813	1389	2138	2104	2152	2000	1875
1509	1714	907	856	948	1897	2149	430471
1203	1721	1384	2121	482	2003	430625	1703
1870	106	669	2044	1463	1686	1777	430030
0	430378	430804	1725	1173	2073	649	0
	2007	974	0	0	486	0	
	1588	0			0		
	0						

Table A.29: Routes of solution E, 50_d1_tw1 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8
0	0	0	0	0	0	0	0
2107	430148	1888	430761	106	1509	1389	1678
1362	2149	1875	2000	2138	2152	1714	1781
1235	1813	430625	2003	430471	1384	907	1721
2104	430030	661	856	2121	1703	1686	1203
948	1870	669	1897	2044	430378	2007	1173
482	0	430804	1777	486	0	1463	0
649		974	1725	0		0	
2073		0	1588				
0			0				

Table A.30: Routes of solution F, 50_d1_tw1 ($Z4$ vs $Z5$).

A.6 50_d1_tw2

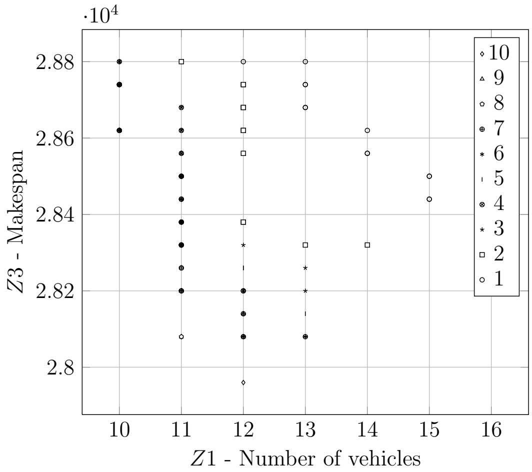

Figure A.61: Front progression of 50_d1_tw2 for $Z1$ vs $Z3$.

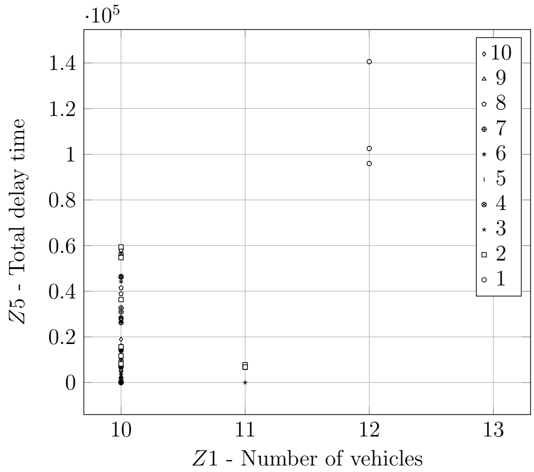

Figure A.62: Front progression of 50_d1_tw2 for $Z1$ vs $Z5$.

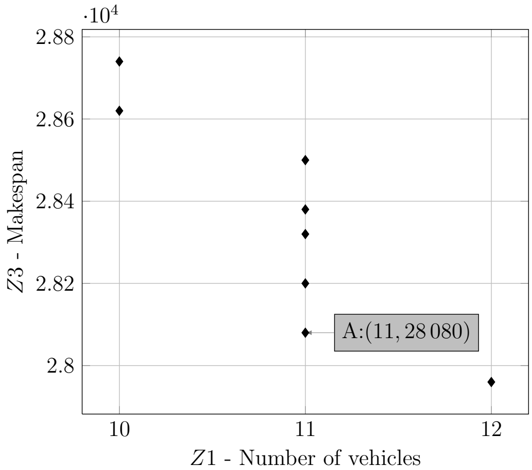

Figure A.63: Final approximation front of 50_d1_tw2 for $Z1$ vs $Z3$.

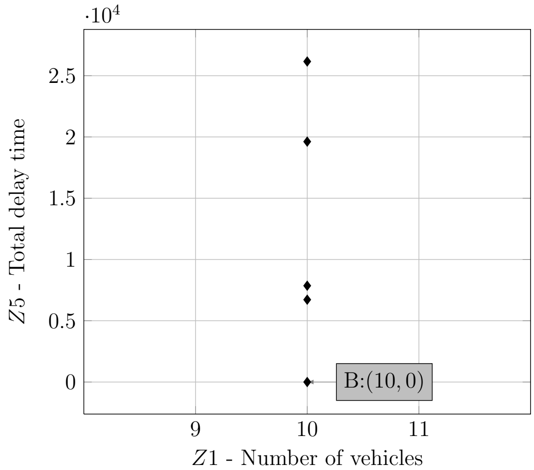

Figure A.64: Final approximation front of 50_d1_tw2 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11
0	0	0	0	0	0	0	0	0	0	0
2003	1888	430761	1714	669	430148	1781	2104	2138	1362	1463
1721	1509	430804	1384	482	430471	1235	856	1203	2000	1725
1813	1897	2044	2152	0	1678	2149	1875	2107	430625	1588
1777	948	0	1870		661	1703	1173	0	1686	0
974	430030		2121		2073	1389	907		649	
0	106				486	0	430378		0	
	0				2007		0			
					0					

Table A.31: Routes of solution A, 50_d1_tw2 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
0	0	0	0	0	0	0	0	0	0
430471	430148	430761	106	2000	2104	1389	1678	2107	1781
1588	1888	1714	661	430625	2003	1235	1362	1703	2138
486	1813	856	1509	669	1203	2149	1721	2044	430030
2007	1897	2152	1875	974	430804	1777	1463	0	0
0	1384	1686	1870	0	0	2121	430378		
	948	1725	482			0	0		
	907	0	0						
	2073	0							
	649								
	1173								
	0								

Table A.32: Routes of solution B, 50_d1_tw2 ($Z1$ vs $Z5$).

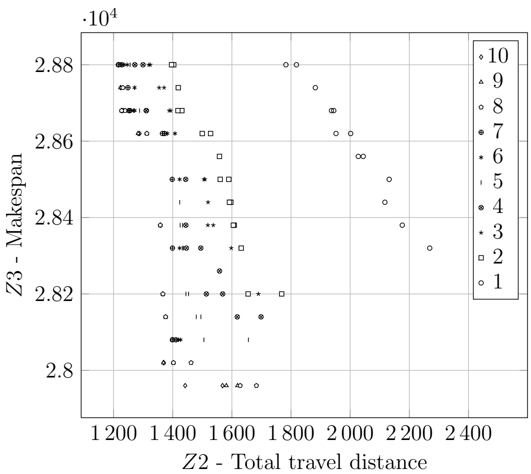

Figure A.65: Front progression of 50_d1_tw2 for $Z2$ vs $Z3$.

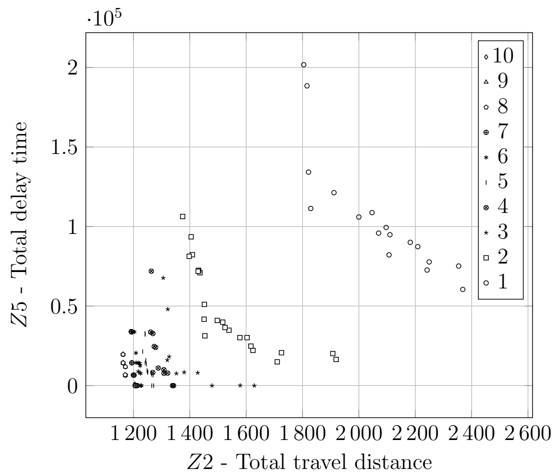

Figure A.66: Front progression of 50_d1_tw2 for $Z2$ vs $Z5$.

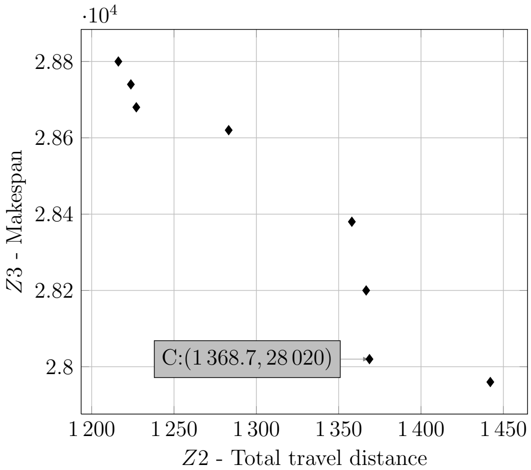

Figure A.67: Final approximation front of 50_d1_tw2 for $Z2$ vs $Z3$.

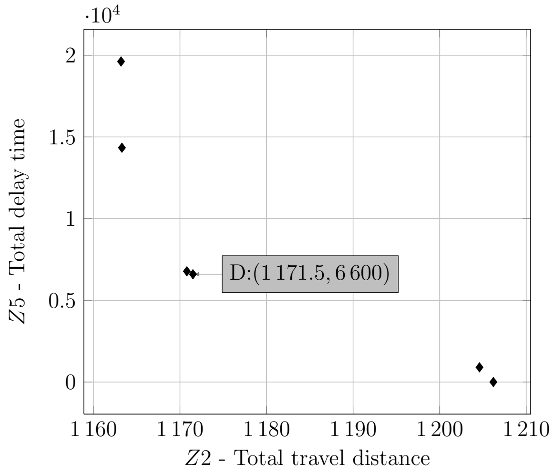

Figure A.68: Final approximation front of 50_d1_tw2 for $Z2$ vs $Z5$.

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**
0	0	0	0	0	0	0	0	0	0	0	0
1721	1384	430761	2149	1870	430471	106	2044	2003	430378	1725	1703
2152	948	2107	1875	482	856	430030	1389	1203	2104	1588	1777
430148	1897	1781	1235	0	661	0	0	2138	1813	1714	0
1686	430625	430804	2121		907			0	0	0	
1888	2073	669	974		1509						
1173	649	0	0		1678						
0	2000				2007						
	1362				486						
	0				1463						
					0						

Table A.33: Routes of solution C, 50_d1_tw2 ($Z2$ vs $Z3$).

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**
0	0	0	0	0	0	0	0	0	0
1678	430471	1870	2104	1714	1888	106	430761	2152	1588
430148	856	482	1813	948	1703	1362	2107	1686	1725
2138	661	0	1897	1384	1777	2000	669	649	0
1203	907		2073	1721	0	2003	430804	0	
974	1509		1173	430625		1781	0		
0	486		430378	430030		2149			
	2007		0	0		1875	0		
	1463					1235			
	0					2044			
						2121			
						1389			
						0			

Table A.34: Routes of solution D, 50_d1_tw2 ($Z2$ vs $Z5$).

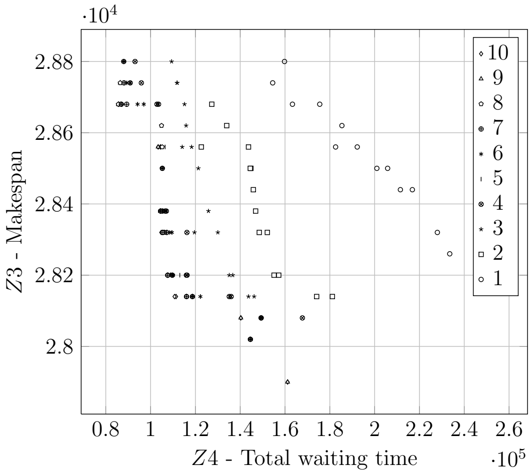

Figure A.69: Front progression of 50_d1_tw2 for $Z4$ vs $Z3$.

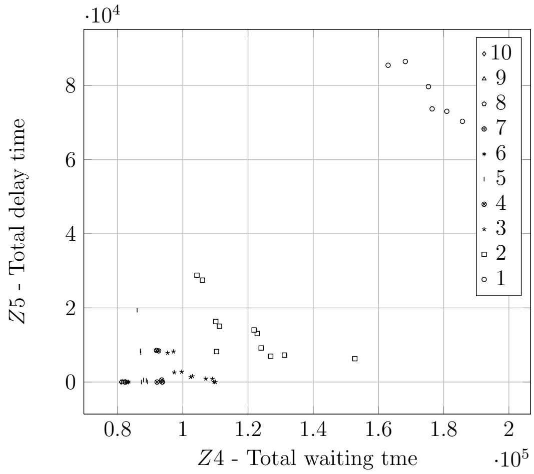

Figure A.70: Front progression of 50_d1_tw2 for $Z4$ vs $Z5$.

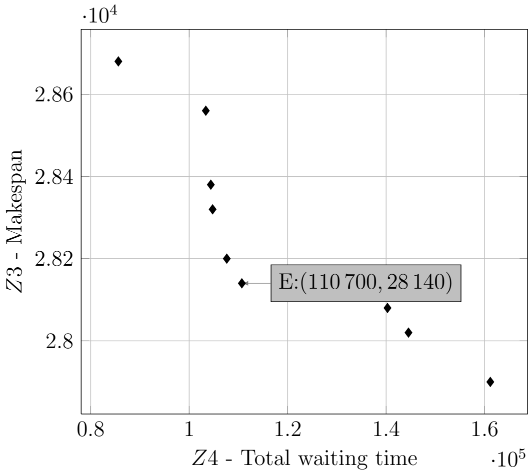

Figure A.71: Final approximation front of 50_d1_tw2 for $Z4$ vs $Z3$.

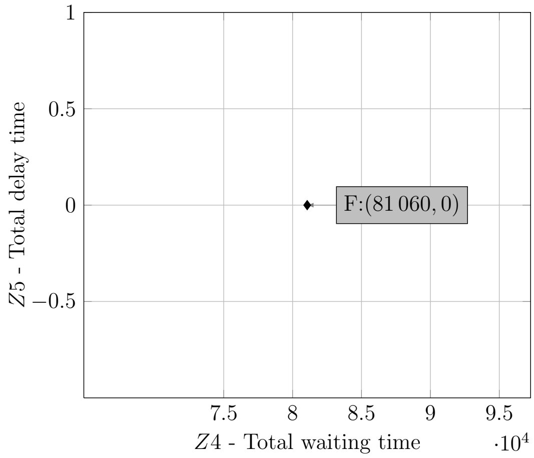

Figure A.72: Final approximation front of 50_d1_tw2 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11
0	0	0	0	0	0	0	0	0	0	0
430148	1714	430761	2138	2107	1781	1813	1235	1389	1875	1888
1897	907	2000	856	1362	2152	1721	1509	2104	948	649
2073	661	2003	106	2149	1777	1384	2121	430030	669	1173
486	1870	1686	1703	430625	430804	430378	974	1678	482	1463
2007	2044	1588	0	430471	0	1725	0	0	0	0
0	0	0		1203		0				
				0						

Table A.35: Routes of solution E, 50_d1_tw2 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
0	0	0	0	0	0	0	0	0	0
1509	430761	2152	2000	1362	106	1721	907	430148	1888
1813	1678	1875	2149	2138	1389	1203	1781	1235	2107
1897	2003	1686	430625	430471	661	482	430378	1384	948
1714	856	1588	2121	1870	1703	0	2007	669	2104
649	1777	0	1725	974	430804		1173	2073	430030
486	2044		0	0	0		0	0	0
1463	0								
0									

Table A.36: Routes of solution F, 50_d1_tw2 ($Z4$ vs $Z5$).

A.7 50_d1_tw3

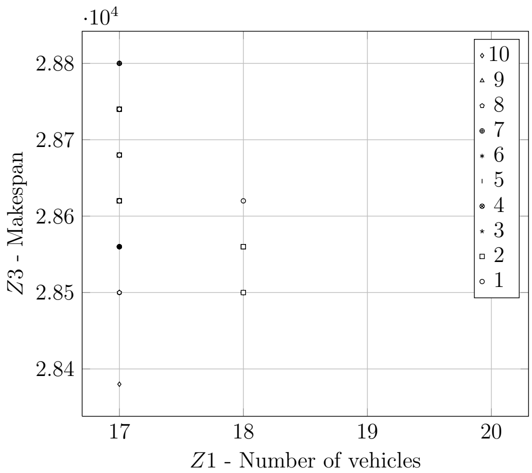

Figure A.73: Front progression of 50_d1_tw3 for $Z1$ vs $Z3$.

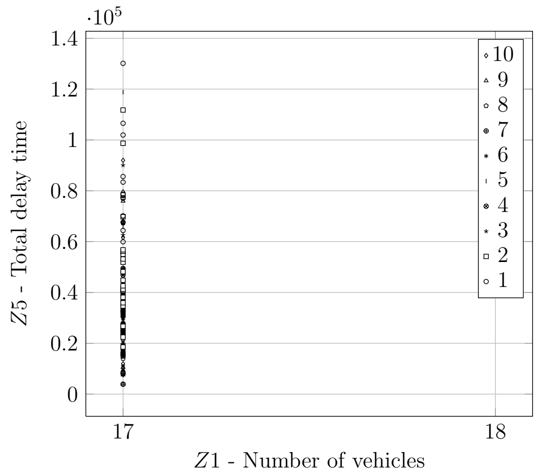

Figure A.74: Front progression of 50_d1_tw3 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
482	1509	2149	2138	1813	1384	907	1875	2000	1389	649	974	2104	430625	2121	2073	430148
2003	1777	1686	1463	1203	948	2007	856	1897	1235	1173	2107	1870	661	0	2152	1888
0	0	106	1588	0	1678	1725	430471	1781	1703	0	0	0	1714		0	430030
		0	0		486	0	1721	669	0				2044			0
					430378		1362	430761					0			
					0		430804	0								
							0									

Table A.37: Routes of solution A, 50_d1_tw3 ($Z1$ vs $Z3$).

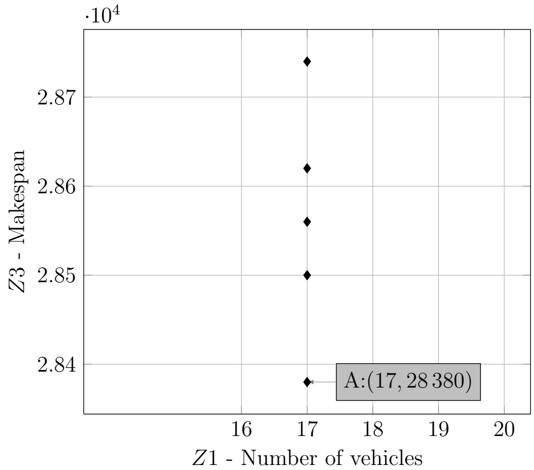

Figure A.75: Final approximation front of 50_d1_tw3 for $Z1$ vs $Z3$.

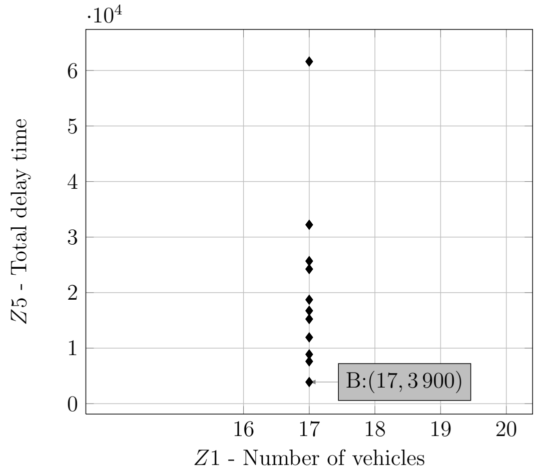

Figure A.76: Final approximation front of 50_d1_tw3 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430148	1714	948	2107	430030	1813	106	1678	2152	1888	2000	2073	430804	2138	1463	482	1686
430471	1703	1897	430761	0	1721	1362	430625	2121	2104	856	430378	0	974	0	0	0
1725	0	1384	1875		2149	2003	661	0	907	2007	0		0			
1173		1235	1389		1781	1509	1203		649	1588						
0		1870	2044		669	1777	0		486	0						
		0	0		0	0			0							

Table A.38: Routes of solution B, 50_d1_tw3 ($Z1$ vs $Z5$).

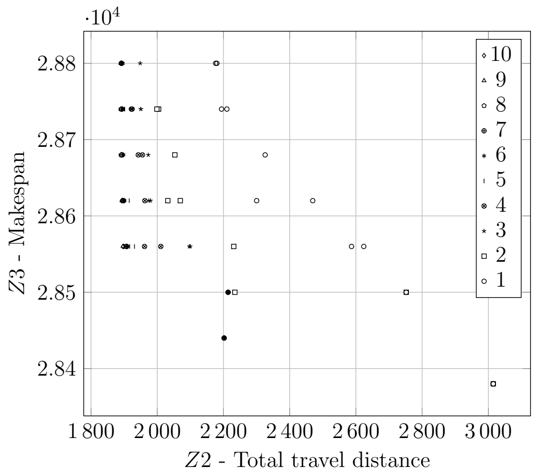

Figure A.77: Front progression of 50_d1_tw3 for $Z2$ vs $Z3$.

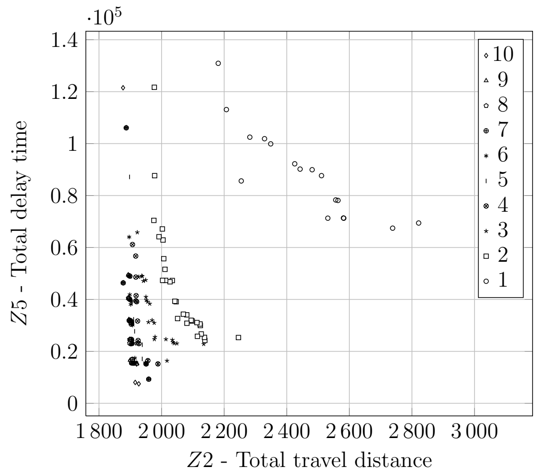

Figure A.78: Front progression of 50_d1_tw3 for $Z2$ vs $Z5$.

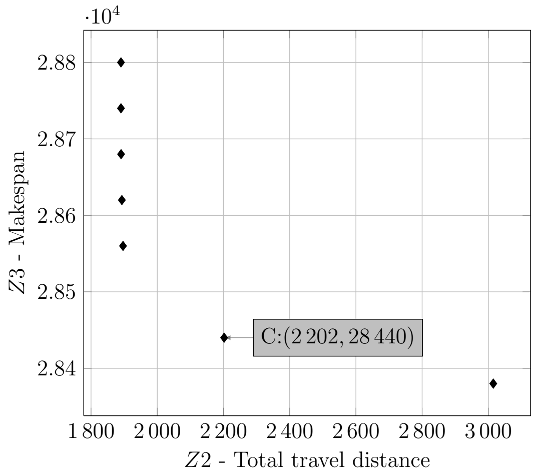

Figure A.79: Final approximation front of 50_d1_tw3 for $Z2$ vs $Z3$.

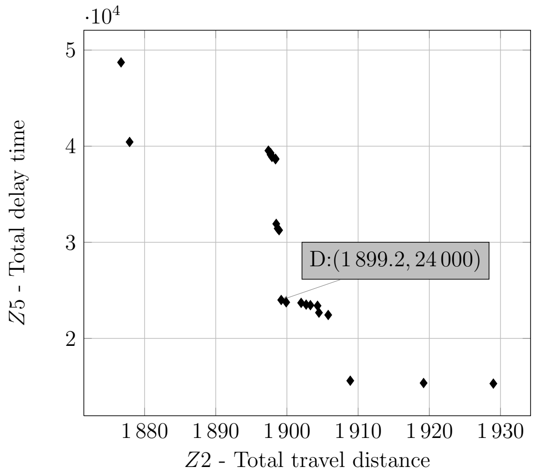

Figure A.80: Final approximation front of 50_d1_tw3 for $Z2$ vs $Z5$.

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**	**V13**	**V14**	**V15**	**V16**	**V17**	**V18**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2107	2003	974	661	2104	1870	430471	1714	430625	482	1173	2121	2152	2149	1703	486	1781	430030
1389	1777	1235	2073	2138	0	2000	430148	1897	0	2007	0	1463	2044	0	1509	430804	0
1875	0	0	649	1203		1721	1686	669		0		430378	0		0	0	0
1588			0	0		1362	1888	430761				0					
1725						1813	0	0									
0						1384											
						948											
						856											
						907											
						1678											
						106											
						0											

Table A.39: Routes of solution C, 50_d1_tw3 ($Z2$ vs $Z3$).

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**	**V13**	**V14**	**V15**	**V16**	**V17**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1813	2107	1389	430471	1870	1235	1362	2000	1714	1725	1777	106	2149	430804	1463	430030	2003
2104	430761	2121	1384	0	2044	430148	1888	430378	1588	0	1678	974	0	0	0	482
856	669	1875	1897		0	1703	2138	486	0		907	0				0
948	1781	0	430625			0	1203	0			661					
2073	0		1721				0				1509					
649			1686								2007					
0			2152								1173					
			0								0					

Table A.40: Routes of solution D, 50_d1_tw3 ($Z2$ vs $Z5$).

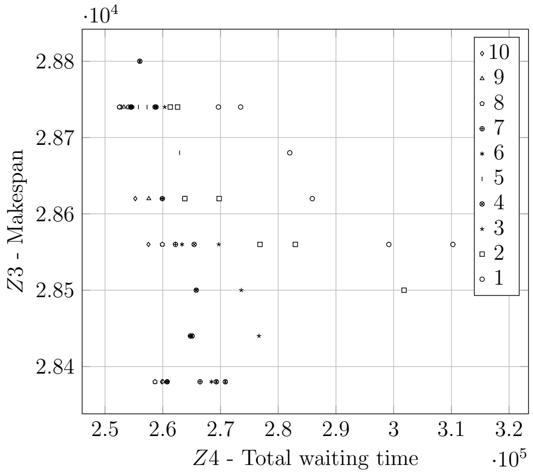

Figure A.81: Front progression of 50_d1_tw3 for $Z4$ vs $Z3$.

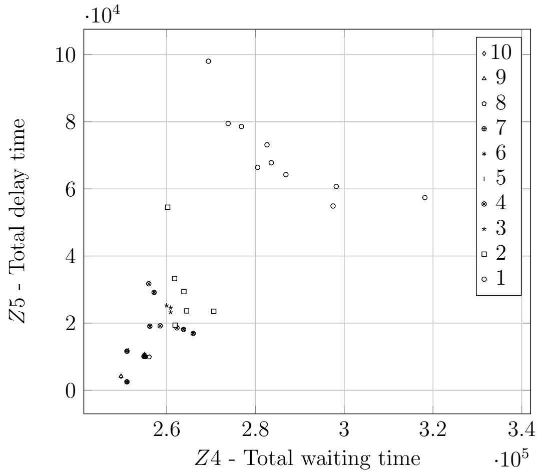

Figure A.82: Front progression of 50_d1_tw3 for $Z4$ vs $Z5$.

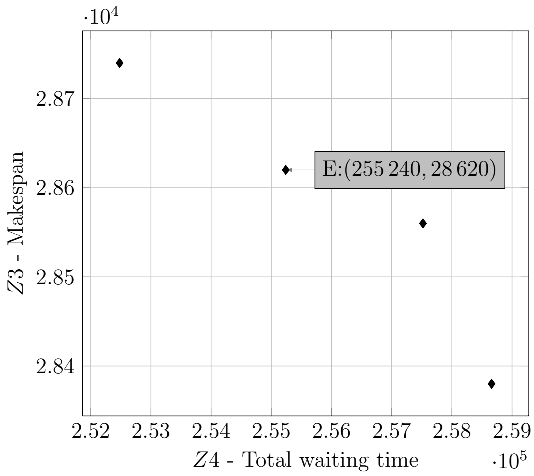

Figure A.83: Final approximation front of 50_d1_tw3 for $Z4$ vs $Z3$.

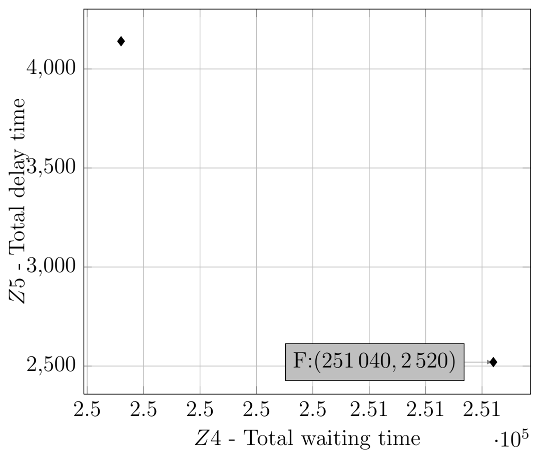

Figure A.84: Final approximation front of 50_d1_tw3 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430148	1781	1235	1678	2003	1203	1173	1875	2107	669	2044	2138	1686	1509	2149	1725	430030
430761	661	856	1588	1721	0	649	430625	1813	0	0	948	1897	907	2104	0	0
1888	1777	430378	2073	482		0	430471	1703			430804	0	2000	2152		
1389	0	486	0	1362			1384	0			106		1870	974		
1714		0		0			2121				0		0	0		
1463							0									
2007																
0																

Table A.41: Routes of solution E, 50_d1_tw3 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1389	1678	430761	430030	106	1875	1888	1777	2121	2044	430148	1725	1362	2138	1509	856	1463
1897	2107	1714	0	2000	1588	1235	0	0	0	2149	2007	430625	1813	430471	1686	0
948	1721	1870		2003	1173	907				1384	0	1203	669	974	0	
430378	1781	0		661	0	2104				430804		0	0	0		
2073	2152			1703		649				0						
0	482			0		486										
	0					0										

Table A.42: Routes of solution F, 50_d1_tw3 ($Z4$ vs $Z5$).

A.8 50_d1_tw4

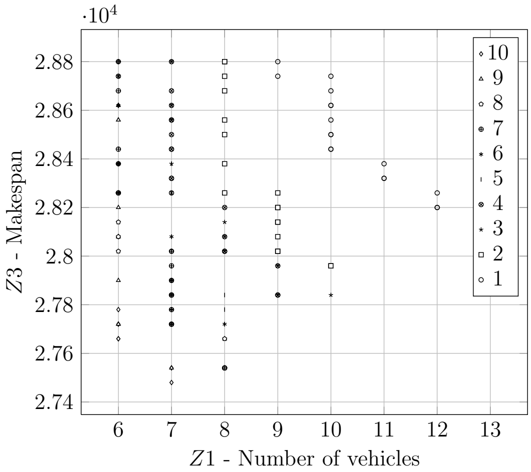

Figure A.85: Front progression of 50_d1_tw4 for $Z1$ vs $Z3$.

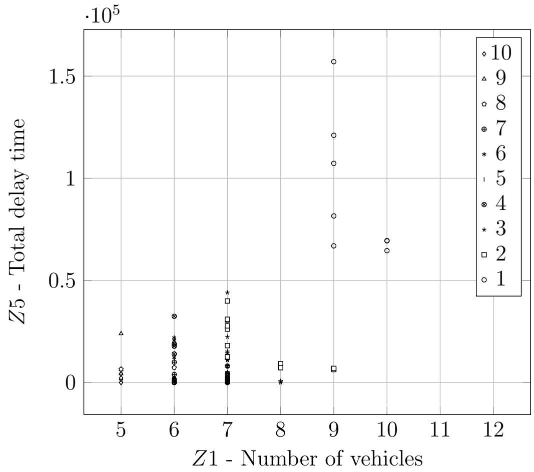

Figure A.86: Front progression of 50_d1_tw4 for $Z1$ vs $Z5$.

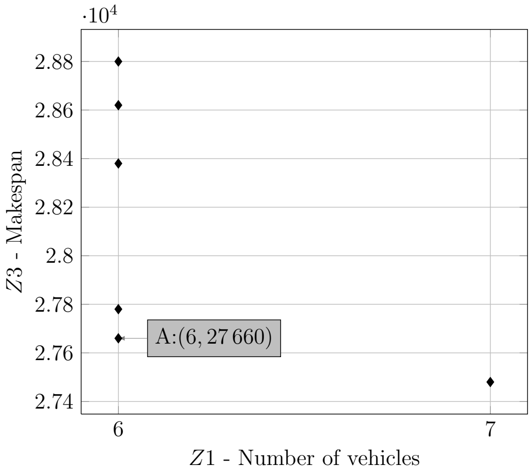

Figure A.87: Final approximation front of 50_d1_tw4 for $Z1$ vs $Z3$.

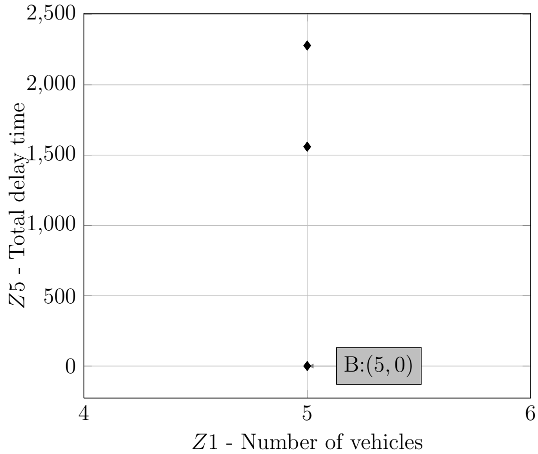

Figure A.88: Final approximation front of 50_d1_tw4 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
649	1725	2000	1173	2104	1870
1384	856	430148	430030	1714	2107
430761	1888	1897	1703	430378	1777
2044	1362	1813	1781	948	1203
2121	1678	1463	669	430625	1389
2149	2073	486	2003	0	1875
430804	1588	1509	0		974
482	1686	661			1235
0	2152	907			2138
	106	430471			0
	0	1721			
		2007			
		0			

Table A.43: Routes of solution A, 50_d1_tw4 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5
0	0	0	0	0
2104	1813	1173	649	1870
856	1725	1384	2107	1777
1888	1897	1714	907	1203
430148	1678	430378	2073	1703
1463	430030	2044	486	1389
1588	974	1875	1509	669
106	430761	1235	1686	482
661	1781	2121	2152	0
430471	2149	2138	1362	
2000	430804	0	1721	
430625	2003		0	
2007	0			
948				
0				

Table A.44: Routes of solution B, 50_d1_tw4 ($Z1$ vs $Z5$).

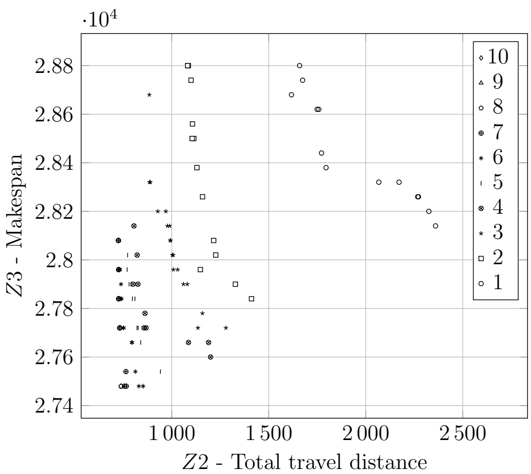

Figure A.89: Front progression of 50_d1_tw4 for $Z2$ vs $Z3$.

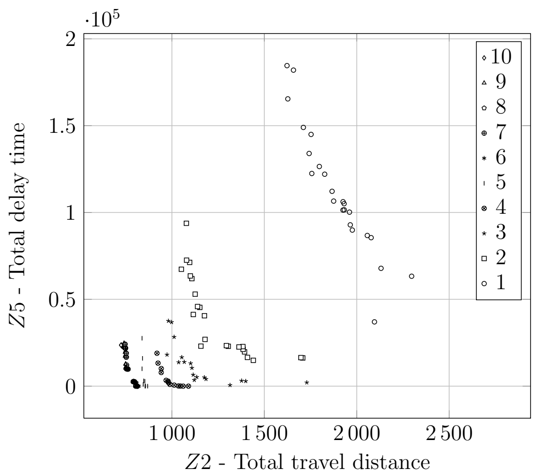

Figure A.90: Front progression of 50_d1_tw4 for $Z2$ vs $Z5$.

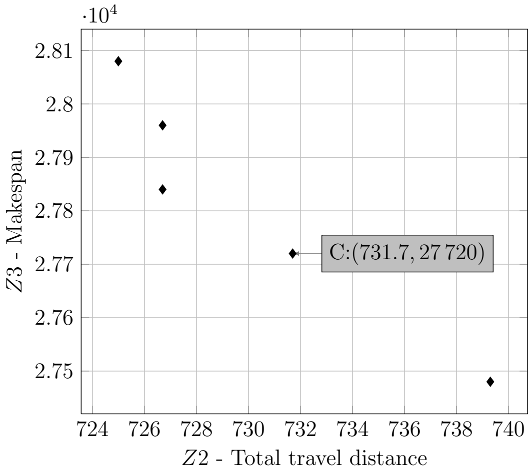

Figure A.91: Final approximation front of 50_d1_tw4 for $Z2$ vs $Z3$.

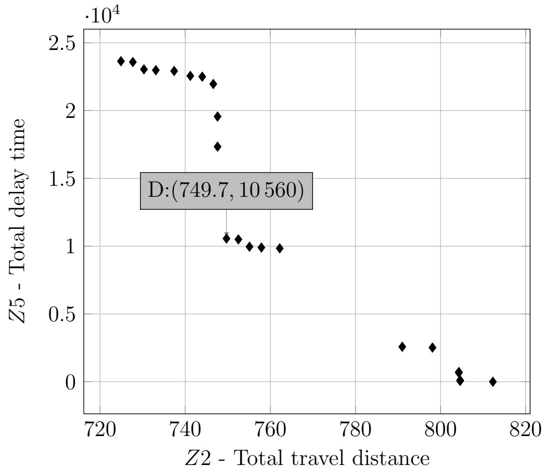

Figure A.92: Final approximation front of 50_d1_tw4 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
1813	1714	1588	430378	430471	1725
2104	649	430761	856	1721	1870
1173	430030	2107	907	2152	1389
1384	1703	974	486	430625	2044
1678	1777	1781	1509	0	2121
661	1203	430804	948		1875
106	2138	482	2007		1235
2073	0	0	1463		2149
1897			0		669
2000					2003
1362					0
1686					
430148					
1888					
0					

Table A.45: Routes of solution C, 50_d1_tw4 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
1725	1870	1714	2104	856	1588
2107	2044	430378	1813	1173	1463
430761	2121	106	1384	649	2073
1781	1389	661	1897	430030	430625
669	1235	486	1888	1703	2000
482	1875	907	430148	1777	0
0	974	1509	1686	1203	
	2149	1678	2152	2138	
	430804	2007	1721	0	
	2003	948	1362		
	0	430471	0		
		0			

Table A.46: Routes of solution D, 50_d1_tw4 ($Z2$ vs $Z5$).

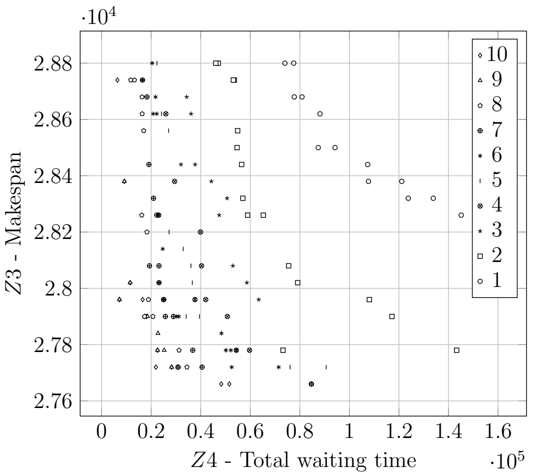

Figure A.93: Front progression of 50_d1_tw4 for $Z4$ vs $Z3$.

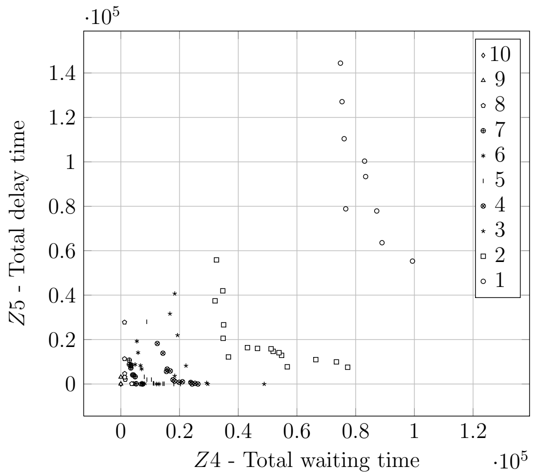

Figure A.94: Front progression of 50_d1_tw4 for $Z4$ vs $Z5$.

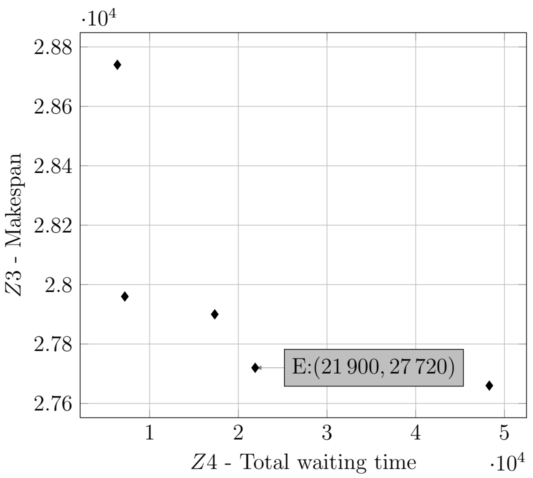

Figure A.95: Final approximation front of 50_d1_tw4 for $Z4$ vs $Z3$.

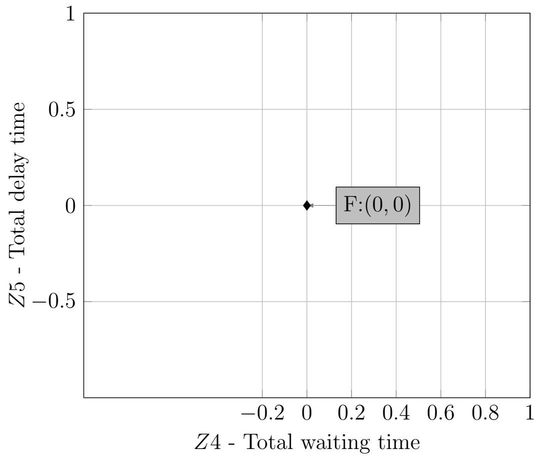

Figure A.96: Final approximation front of 50_d1_tw4 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
2107	2104	1725	1777	649	1870
1897	430378	1888	2000	1714	430030
1703	2121	974	2044	1678	1389
1235	907	1588	1686	1362	1509
2149	856	1781	2152	1384	1721
430804	669	2138	106	486	430625
482	2003	0	0	2073	430471
0	0			948	0
				661	
				2007	
				1463	
				0	

Table A.47: Routes of solution E, 50_d1_tw4 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6
0	0	0	0	0	0
649	2107	1725	1870	1897	1173
430378	1777	1384	2000	1813	856
1203	106	2104	1875	1714	1888
661	1463	1703	2044	430030	430761
2149	1509	1588	2121	974	1235
430471	907	669	1686	1389	2073
430625	486	2003	2152	1721	430148
0	1781	0	1362	2007	2138
	430804		0	948	0
	482			1678	
	0			0	

Table A.48: Routes of solution F, 50_d1_tw4 ($Z4$ vs $Z5$).

A.9 50_d2_tw1

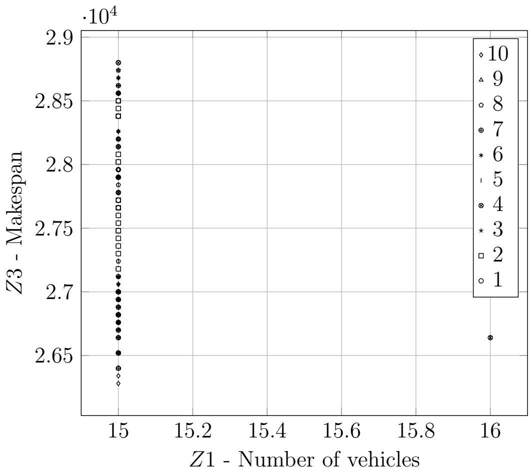

Figure A.97: Front progression of 50_d2_tw1 for $Z1$ vs $Z3$.

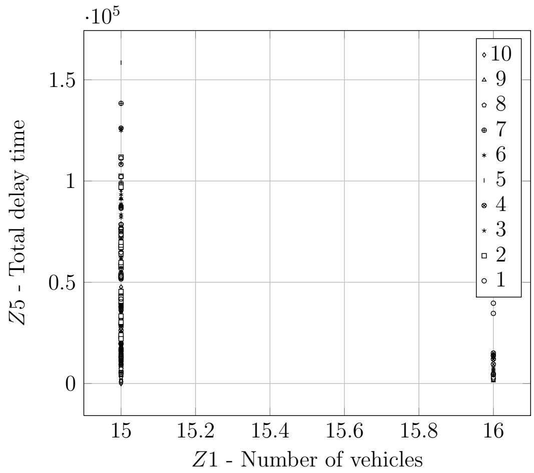

Figure A.98: Front progression of 50_d2_tw1 for $Z1$ vs $Z5$.

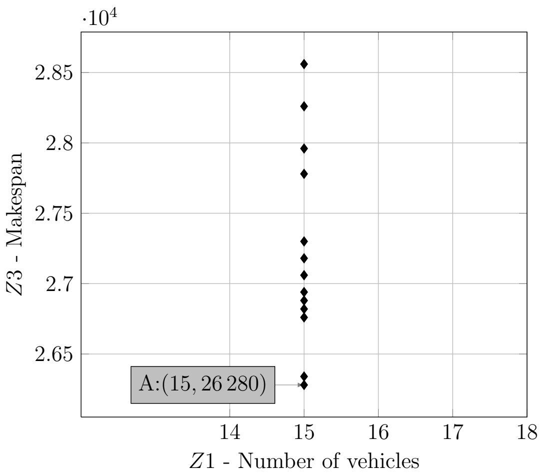

Figure A.99: Final approximation front of 50_d2_tw1 for $Z1$ vs $Z3$.

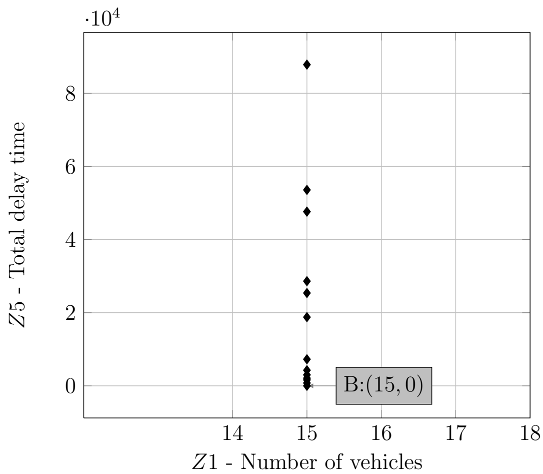

Figure A.100: Final approximation front of 50_d2_tw1 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1721	430625	2149	1781	2003	661	2152	1897	1389	1235	1875	856	1384	1813	1203
948	2044	430471	1463	1870	2138	1173	482	974	2007	1686	649	669	1714	0
430148	1588	430030	2073	430804	1777	907	430378	2121	1509	2104	486	1725	1703	
1888	0	106	0	0	1678	1362	0	0	2000	0	0	0	0	
430761		0			0	0			0					
2107														
0														

Table A.49: Routes of solution A, 50_d2_tw1 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1235	430148	2138	2149	856	1203	1721	948	430471	430030	1897	1875	1714	1588	430761
669	1888	1703	1686	907	430378	2104	2003	482	1777	2073	1870	649	1725	2107
1173	106	974	1463	661	0	1813	1384	486	0	2007	430804	2044	0	1389
0	1678	0	0	2152		1509	430625	0		0	0	0		2121
	2000			0		0	0							0
	1362													
	1781													

Table A.50: Routes of solution B, 50_d2_tw1 ($Z1$ vs $Z5$).

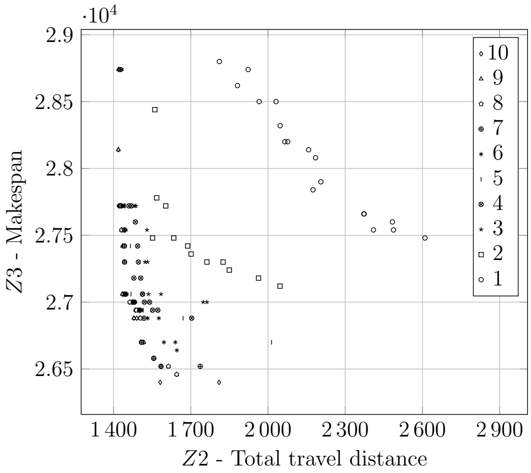

Figure A.101: Front progression of 50_d2_tw1 for $Z2$ vs $Z3$.

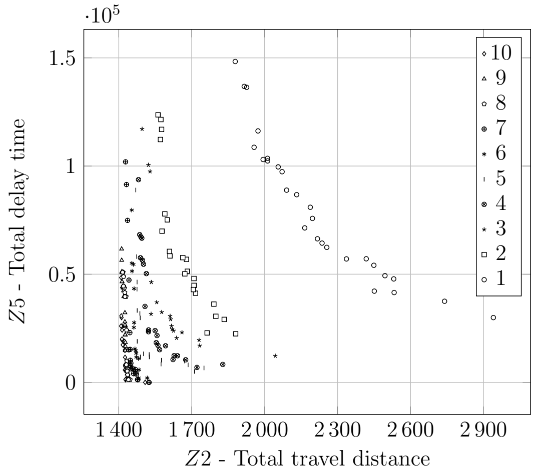

Figure A.102: Front progression of 50_d2_tw1 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1588	1714	649	1875	2152	1235	907	2138	1389	486	2003	2073	1173	482	430030
1725	430378	430625	2149	1686	669	661	1203	2121	2007	1870	1897	2104	0	1703
1813	430471	1384	430761	430148	430804	856	1777	2044	948	974	1721	1463		0
0	106	1678	1781	1888	0	1509	0	0	0	0	2000	0		
	0	0	2107	1362		0					0			
			0	0										

Table A.51: Routes of solution C, 50_d2_tw1 ($Z2$ vs $Z3$).

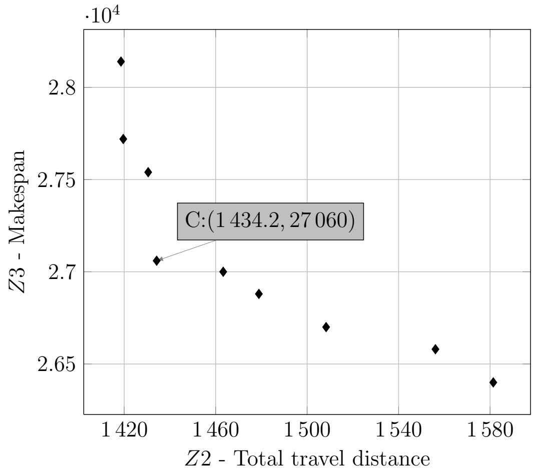

Figure A.103: Final approximation front of 50_d2_tw1 for $Z2$ vs $Z3$.

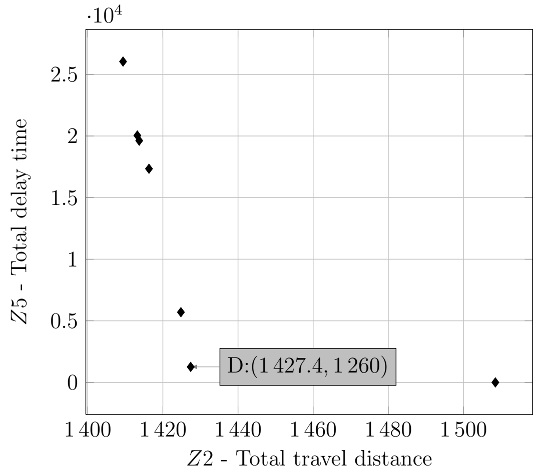

Figure A.104: Final approximation front of 50_d2_tw1 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430761	106	1813	2138	948	1714	430030	2152	1389	1509	430471	1781	2003	2121	486
2107	1678	2104	1203	1384	430378	1703	1686	2044	661	1173	2149	482	1870	2007
430804	430148	1588	1777	430625	1463	0	649	974	856	2073	1875	1725	0	0
669	1888	0	0	1897	0		0	0	907	0	1235	0		
0	2000			0					0		0			
	1721													
	1362													
	0													

Table A.52: Routes of solution D, 50_d1_tw1 ($Z2$ vs $Z5$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
106	1384	2000	430148	1509	2138	2107	1389	856	2149	1362	1875	430761	2152	1678
1235	1725	1888	1781	669	430378	430471	1173	974	649	2104	1686	1714	2121	0
1897	486	907	1813	430804	1463	2003	1588	482	2007	1721	2073	948	2044	
1870	0	430625	1203	0	0	430030	0	0	0	1777	0	1703	0	
0		661	0			0				0		0		
		0												

Table A.53: Routes of solution E, 50_d2_tw1 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1714	430761	1781	1721	430148	2104	1678	661	1875	2138	1362	1389	1235	2152	106
430030	1888	430378	1870	2149	2044	2000	1777	486	649	2107	1686	2007	1203	0
1725	1813	1173	1463	430471	2121	2003	430804	2073	1588	907	482	669	974	
0	856	0	0	1703	0	948	0	0	0	430625	0	0	0	
	1384			0		1897				1509				
	0					0				0				

Table A.54: Routes of solution F, 50_d2_tw1 ($Z4$ vs $Z5$).

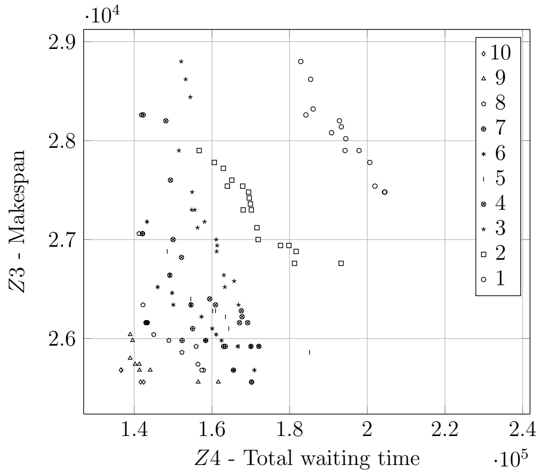

Figure A.105: Front progression of 50_d2_tw1 for $Z4$ vs $Z3$.

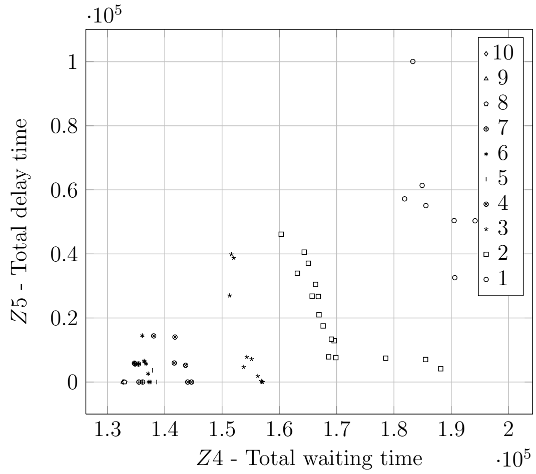

Figure A.106: Front progression of 50_d2_tw1 for $Z4$ vs $Z5$.

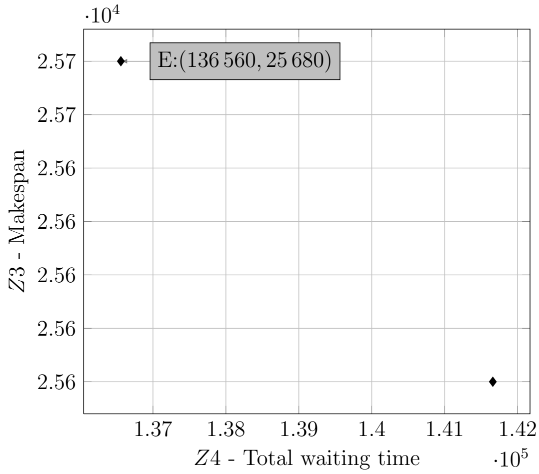

Figure A.107: Final approximation front of 50_d2_tw1 for $Z4$ vs $Z3$.

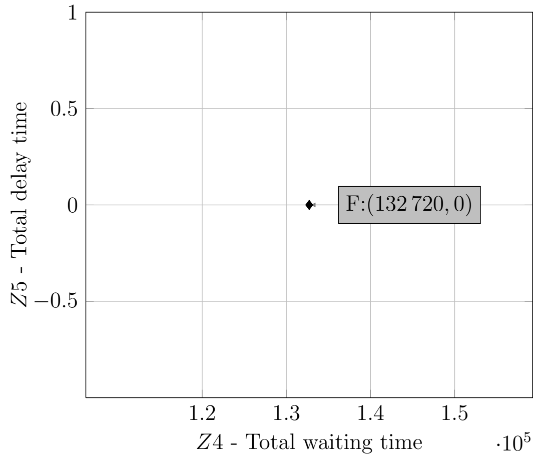

Figure A.108: Final approximation front of 50_d2_tw1 for $Z4$ vs $Z5$.

A.10 50_d2_tw2

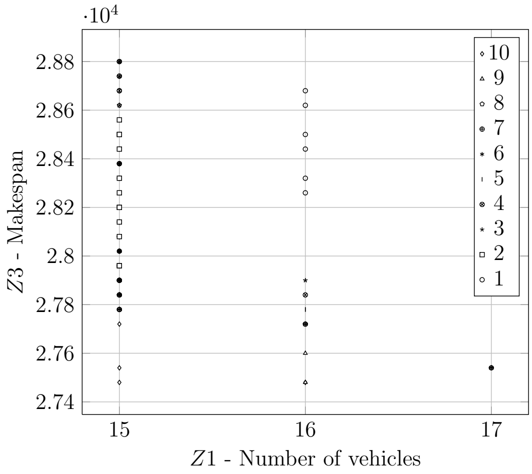

Figure A.109: Front progression of 50_d2_tw2 for $Z1$ vs $Z3$.

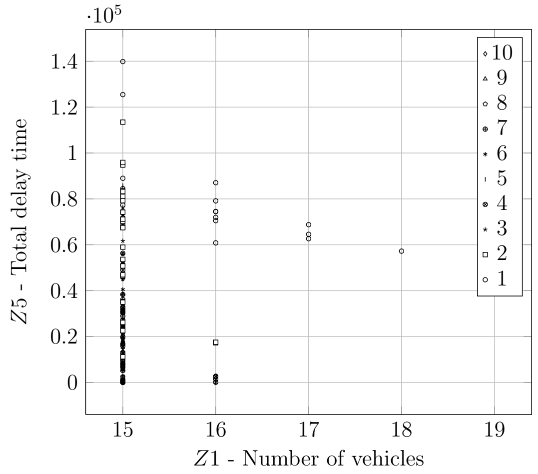

Figure A.110: Front progression of 50_d2_tw2 for $Z1$ vs $Z5$.

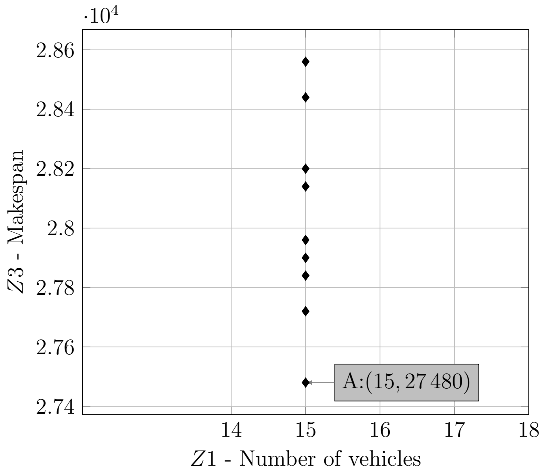

Figure A.111: Final approximation front of 50_d2_tw2 for $Z1$ vs $Z3$.

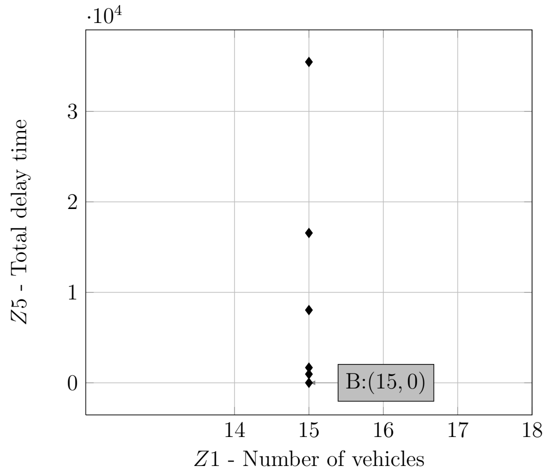

Figure A.112: Final approximation front of 50_d2_tw2 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1389	430625	1509	1813	1714	1721	430471	907	1875	1781	856	2152	2149	1897	2073
2107	1888	430148	1588	1686	1725	430378	649	1235	669	2121	974	2138	486	0
2003	1362	661	106	1463	1173	2007	1384	1870	430804	2044	482	1203	948	
1777	2000	430030	2104	0	0	0	1678	430761	0	0	0	0	0	
0	1703	0	0				0	0						
	0													

Table A.55: Routes of solution A, 50_d2_tw2 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
106	856	430148	2107	1389	1384	1813	2138	2149	1721	2104	2152	1714	430804	2003
1888	1173	1686	430761	1203	649	948	2073	1875	430625	1777	1897	2007	2121	669
1362	430378	486	1870	974	1463	1703	1725	1235	907	1588	1509	482	0	0
2000	0	0	2044	0	0	0	0	661	1781	0	430471	0		
1678			0					0	0		0			
430030														
0														

Table A.56: Routes of solution B, 50_d2_tw2 ($Z1$ vs $Z5$).

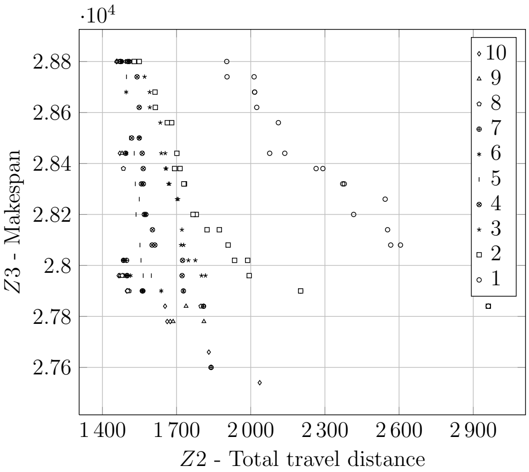

Figure A.113: Front progression of 50_d2_tw2 for $Z2$ vs $Z3$.

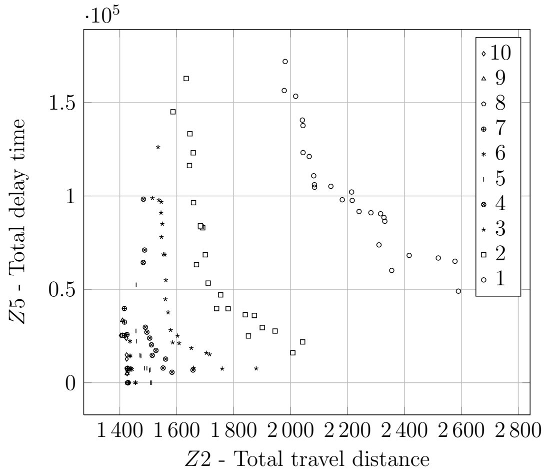

Figure A.114: Front progression of 50_d2_tw2 for $Z2$ vs $Z5$.

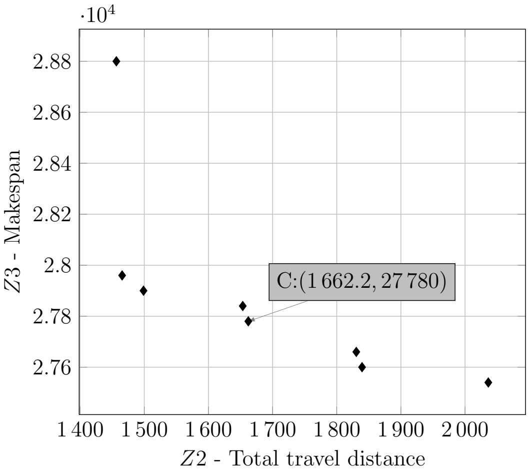

Figure A.115: Final approximation front of 50_d2_tw2 for $Z2$ vs $Z3$.

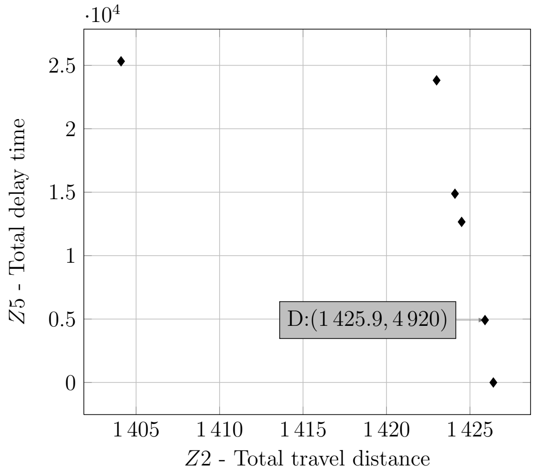

Figure A.116: Final approximation front of 50_d2_tw2 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
486	1781	1389	1173	1875	430471	1813	1721	2152	2003	1235	1714	1897	856	1777
948	2107	2121	1384	430804	430378	2104	649	1686	669	430761	1588	1509	907	0
2007	2149	2044	2073	974	106	430030	430625	430148	482	2138	1725	1463	1870	0
0	1703	0	0	0	661	1362	2000	1888	0	1203	0	0	0	0
	0				0	0	0	1678		0				
								0						

Table A.57: Routes of solution C, 50_d2_tw2 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430761	948	2152	856	1509	430378	1813	2003	1389	1781	106	2138	1888	2000	1714
2107	1384	1686	486	907	430471	1725	1870	2121	2149	1678	1203	430148	1362	0
669	430625	649	2007	661	1463	1588	482	2044	1875	1173	974	1703	1721	0
430804	1897	0	0	2104	0	0	0	0	1235	2073	0	1777	430030	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A.58: Routes of solution D, 50_d2_tw2 ($Z2$ vs $Z5$).

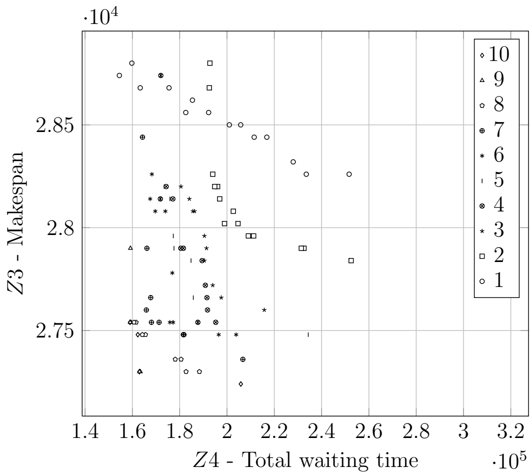

Figure A.117: Front progression of 50_d2_tw2 for $Z4$ vs $Z3$.

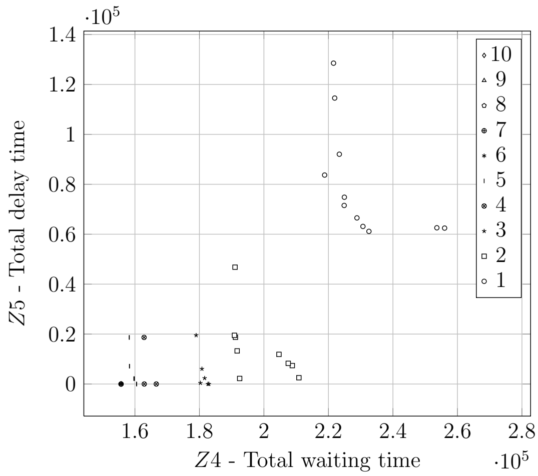

Figure A.118: Front progression of 50_d2_tw2 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2107	1362	2138	430471	1389	1888	1509	2003	430148	1235	661	1875	430761	2152	1678
948	1781	486	2121	1686	856	106	2007	2149	2073	1813	649	1384	430804	0
1714	1897	1725	2044	1463	430625	907	1588	1721	430378	2000	1173	2104	974	0
1777	669	0	0	0	1870	482	0	1703	0	1203	0	430030	0	0
0	0				0	0		0		0		0		

Table A.59: Routes of solution E, 50_d2_tw2 ($Z4$ vs $Z3$).

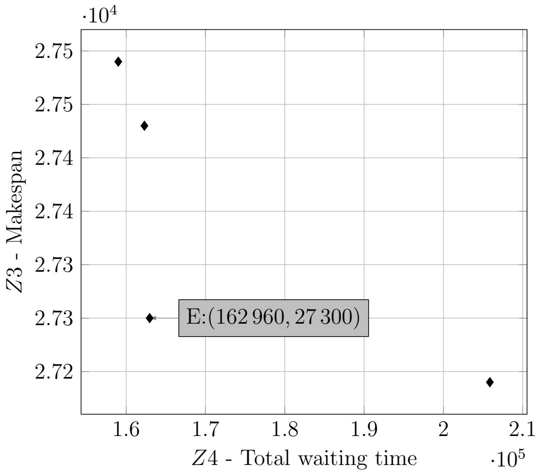

Figure A.119: Final approximation front of 50_d2_tw2 for $Z4$ vs $Z3$.

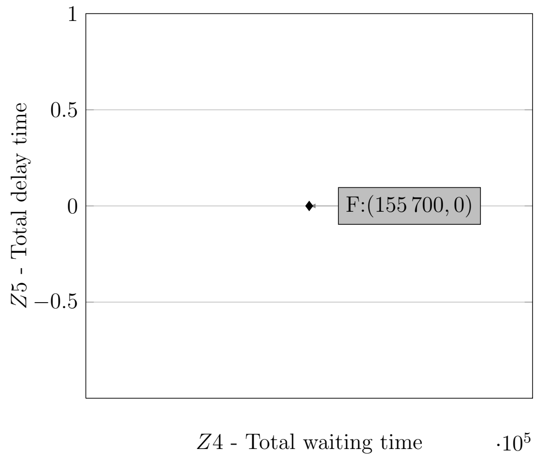

Figure A.120: Final approximation front of 50_d2_tw2 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430625	1678	1875	2152	1235	1781	2138	1362	661	1714	1721	1888	1389	2000	106
2044	430148	2073	1870	1686	649	2007	430761	1203	1703	1777	2107	430378	2149	0
974	1509	486	2121	1588	1463	1725	430471	482	430804	669	1897	1173	2104	
0	1813	0	0	0	0	0	856	0	0	0	1384	0	430030	
	2003						907				948		0	
	0						0				0			

Table A.60: Routes of solution F, 50_d2_tw2 ($Z4$ vs $Z5$).

A.11 50_d2_tw3

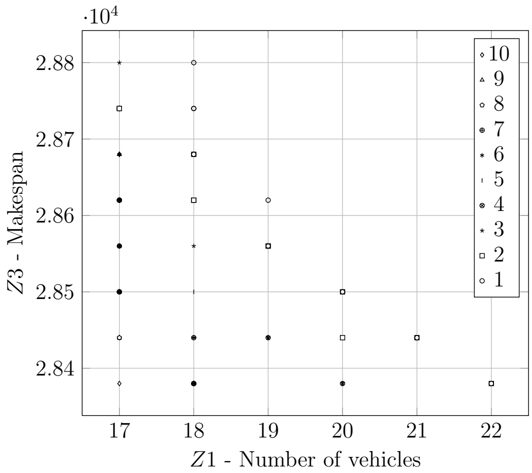

Figure A.121: Front progression of 50_d2_tw3 for $Z1$ vs $Z3$.

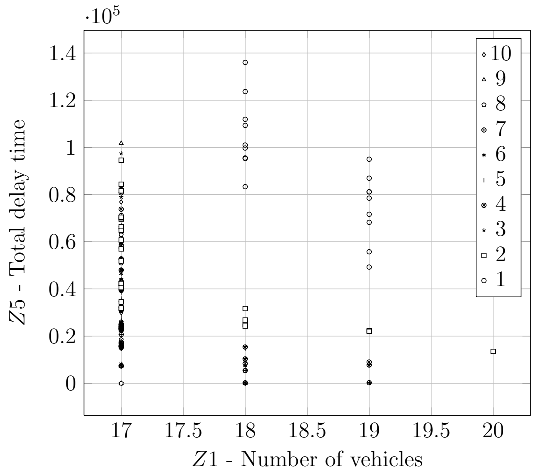

Figure A.122: Front progression of 50_d2_tw3 for $Z1$ vs $Z5$.

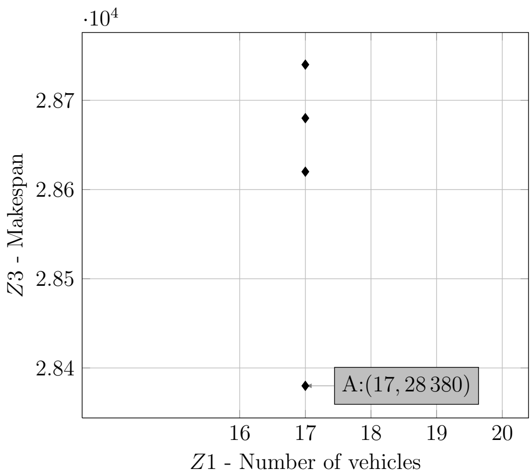

Figure A.123: Final approximation front of 50_d2_tw3 for $Z1$ vs $Z3$.

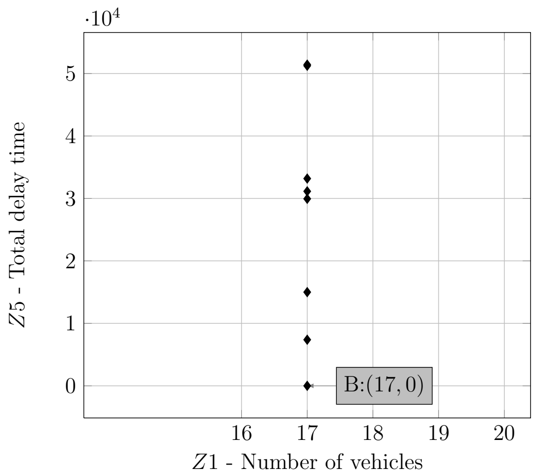

Figure A.124: Final approximation front of 50_d2_tw3 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
856	1813	1714	1721	1235	1875	948	1384	2003	907	1389	430471	2149	1509	2138	430148	430030
2007	1781	1725	1203	1870	1888	2000	1173	486	430804	2044	430378	2104	1362	1897	661	0
2073	974	1463	0	0	1686	1777	649	1588	430761	0	430625	2121	106	482	2152	
0	2107	0			1678	0	0	0	0		0	0	669	0	1703	
	0				0								0		0	

Table A.61: Routes of solution A, 50_d2_tw3 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2000	907	2107	430148	948	430761	1678	1721	106	856	1813	974	1362	1384	1888	1389	661
2138	430471	1235	2104	1509	1781	1588	1703	2149	430378	2073	0	1714	2003	2007	2044	1173
2152	430030	1875	1897	1203	1870	1725	0	669	486	1463		430625	1777	649	0	0
2121	0	1686	430804	0	0	0	0	0	0	0		482	0	0		
0		0	0									0				

Table A.62: Routes of solution B, 50_d2_tw3 ($Z1$ vs $Z5$).

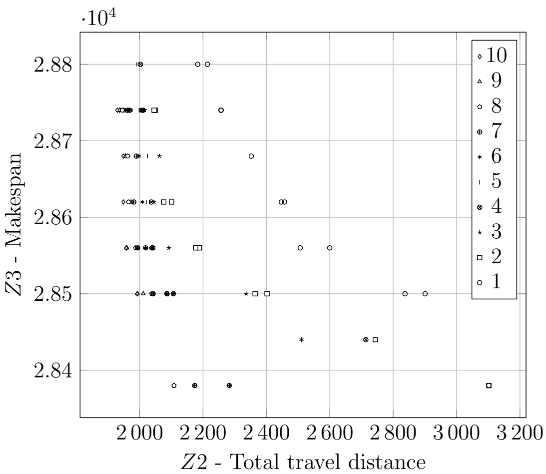

Figure A.125: Front progression of 50_d2_tw3 for $Z2$ vs $Z3$.

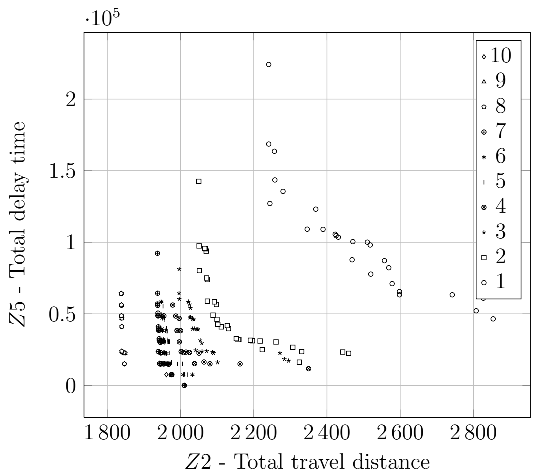

Figure A.126: Front progression of 50_d2_tw3 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1781	1384	1714	1725	1721	1875	1897	856	482	1509	430625	2104	2149	1870	1389	2044	430471	1777	430030
430804	1362	1463	1588	649	669	1888	907	2003	1173	1686	486	974	0	2121	0	430378	0	0
430761	2000	1813	0	2073	2107	2138	661	0	948	2152	2007	1235		0		0		
0	430148	0		0	0	1203	106		0	0	0	0						
	1703					0	1678											
	0						0											

Table A.63: Routes of solution C, 50_d2_tw3 ($Z2$ vs $Z3$).

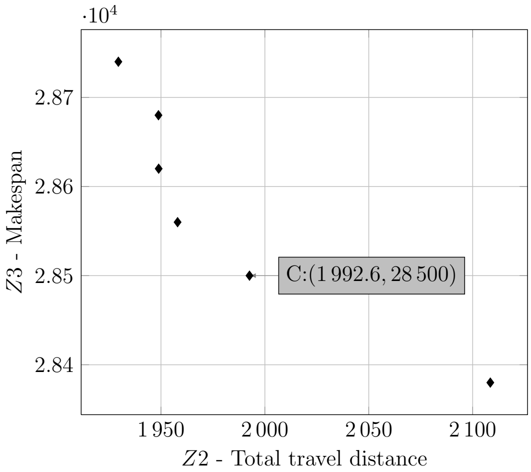

Figure A.127: Final approximation front of 50_d2_tw3 for $Z2$ vs $Z3$.

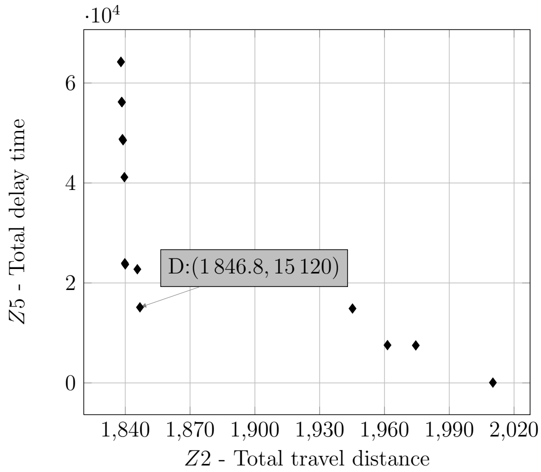

Figure A.128: Final approximation front of 50_d2_tw3 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1870	2000	2149	430625	1725	106	948	1813	430471	430804	1389	2107	1714	1235	1509	1384	2138	974
0	2152	1777	1897	1588	1678	2073	2104	856	1781	2121	430761	2003	2044	2007	1173	1203	0
	1721	0	430030	0	1362	649	1463	661	0	1875	669	482	0	486	430378	0	
	1686		0		1888	0	0	907		0	0	0		0	0		
	0				430148			0									
					1703												
					0												

Table A.64: Routes of solution D, 50_d2_tw3 ($Z2$ vs $Z5$).

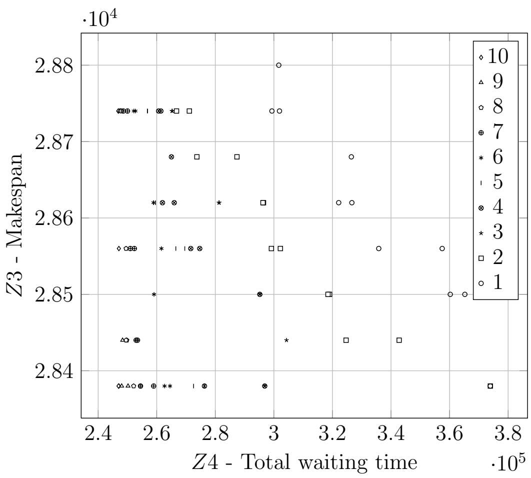

Figure A.129: Front progression of 50_d2_tw3 for $Z4$ vs $Z3$.

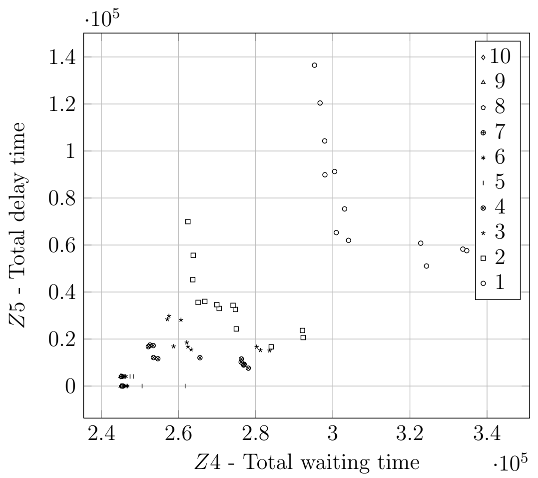

Figure A.130: Front progression of 50_d2_tw3 for $Z4$ vs $Z5$.

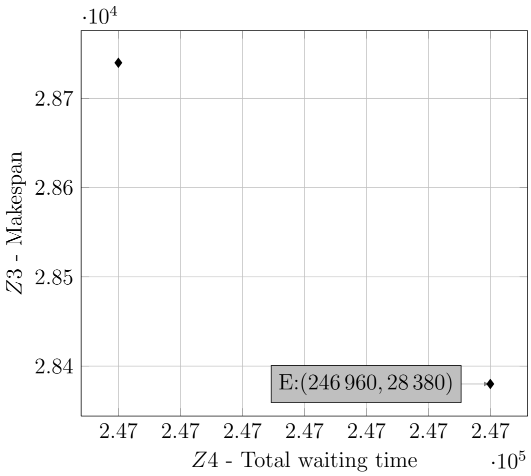

Figure A.131: Final approximation front of 50_d2_tw3 for $Z4$ vs $Z3$.

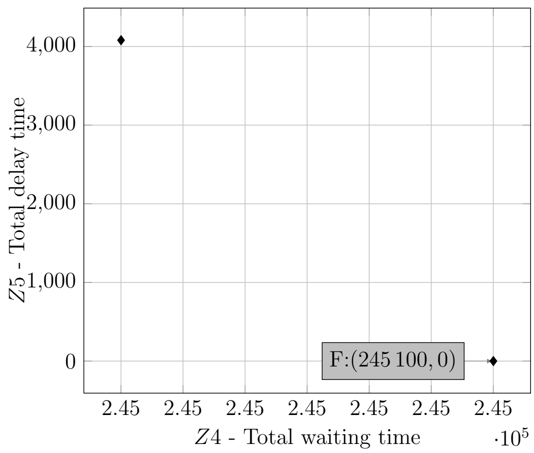

Figure A.132: Final approximation front of 50_d2_tw3 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1875	1384	2000	1235	2138	1389	661	1781	1509	2073	2152	1813	1721	2149	948	2003	430030
856	1777	1703	1888	1897	649	1588	2104	1203	1463	2107	430148	430761	430378	669	430625	0
1870	0	0	1714	106	2007	1173	1678	0	0	1362	430471	907	486	0	2121	
0			1686	974	0	0	1725			2044	482	430804	0		0	
			0	0			0			0	0	0				

Table A.65: Routes of solution E, 50_d2_tw3 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1362	2107	1678	1888	1875	661	2000	430030	2007	106	1384	1897	430148	2149	649	2138	1686
2003	1813	1389	1235	856	1870	1714	0	1588	430761	1725	1703	1781	1721	1463	430378	0
948	907	430471	430625	1777	0	2152		0	2104	1173	0	1509	2044	0	486	
430804	1203	669	2121	0		482			2073	0		974	0		0	
0	0	0	0			0			0			0				

Table A.66: Routes of solution F, 50_d2_tw3 ($Z4$ vs $Z5$).

A.12 50_d2_tw4

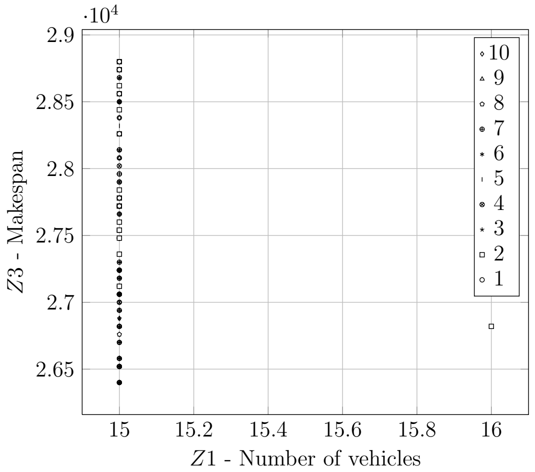

Figure A.133: Front progression of 50_d2_tw4 for $Z1$ vs $Z3$.

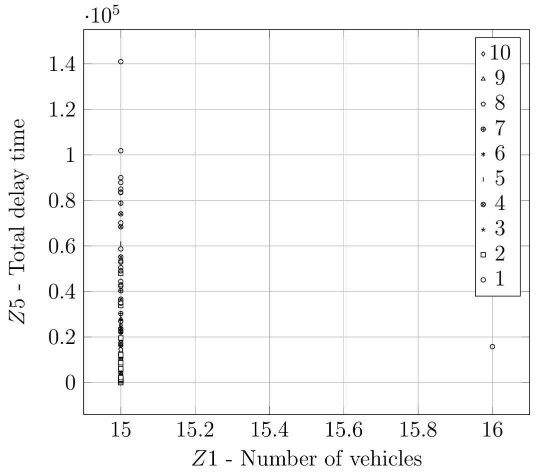

Figure A.134: Front progression of 50_d2_tw4 for $Z1$ vs $Z5$.

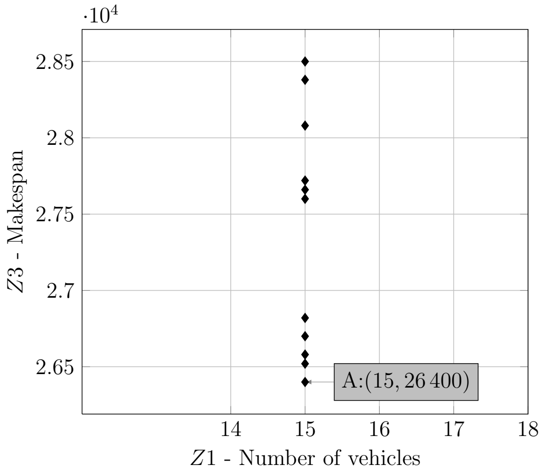

Figure A.135: Final approximation front of 50_d2_tw4 for $Z1$ vs $Z3$.

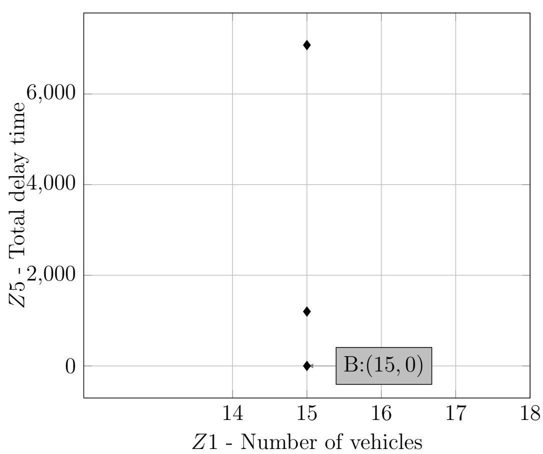

Figure A.136: Final approximation front of 50_d2_tw4 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
661	2073	2104	1777	1203	486	1875	1703	2121	1588	2044	430378	1813	974	2138
1725	1384	1678	907	1870	948	649	2149	430030	1173	1235	1781	1721	430471	0
2007	1389	106	1714	2152	482	669	430804	1509	2003	1686	430761	856	1463	
0	2107	430148	1888	0	0	0	0	0	0	0	2000	1897	0	
	0	1362	0								0	0		
		430625												
		0												

Table A.67: Routes of solution A, 50_d1_tw4 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430030	2104	1384	1588	430378	1813	1203	1777	1897	2073	1870	1173	649	1463	2003
974	856	1725	907	1714	1888	1509	661	486	669	2044	2107	1389	2121	0
2138	482	1721	430761	1703	430804	2149	2007	1235	948	1875	1686	2152	430471	0
0	1362	2000	1781	0	430625	430148	0	106	0	0	1678	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A.68: Routes of solution B, 50_d1_tw4 ($Z1$ vs $Z5$).

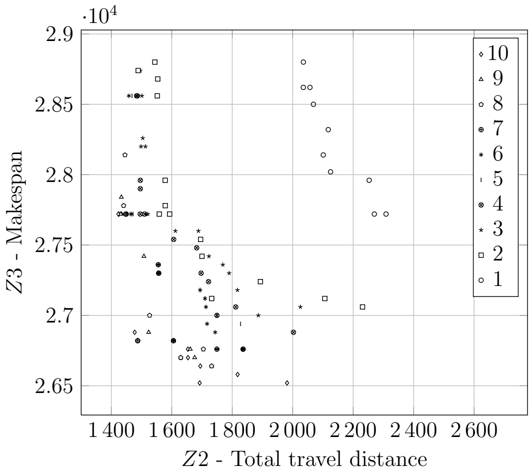

Figure A.137: Front progression of 50_d2_tw4 for $Z2$ vs $Z3$.

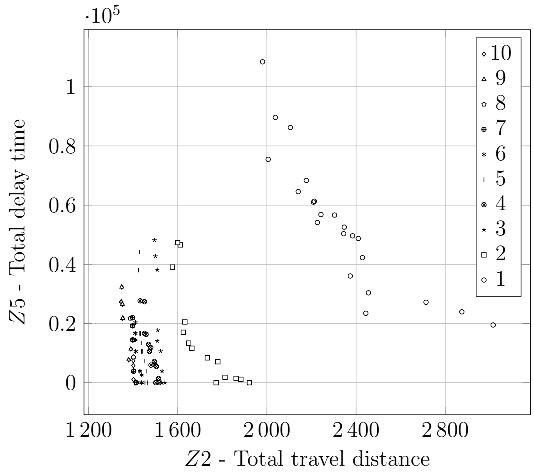

Figure A.138: Front progression of 50_d2_tw4 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1777	1813	1173	1875	974	649	2104	856	430378	1714	1725	2121	2044	430471	430030
1703	1463	948	2149	430804	1686	1509	907	486	2073	482	1389	1870	0	1203
430148	1588	1362	669	1235	2152	2007	106	1897	430625	2003	1781	661		2138
1888	0	1721	2107	0	0	1678	1384	0	0	0	430761	0		0
0		0	0			0	2000				0			
							0							

Table A.69: Routes of solution C, 50_d2_tw4 ($Z2$ vs $Z3$).

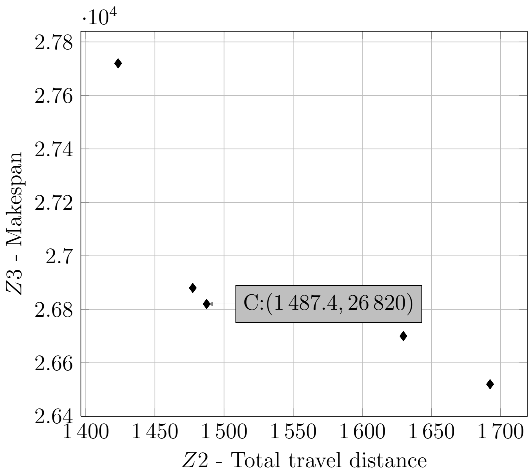

Figure A.139: Final approximation front of 50_d2_tw4 for $Z2$ vs $Z3$.

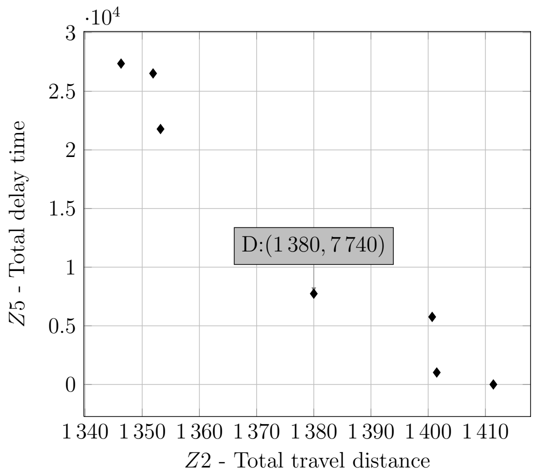

Figure A.140: Final approximation front of 50_d2_tw4 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
974	1813	2107	2044	1173	430378	1686	1725	430471	1888	1588	1714	1777	1870	2073
430804	2104	430761	2121	1897	486	2152	482	948	430030	1463	1384	1203	669	0
1781	907	2149	1389	856	1509	1721	2003	1678	1703	430625	649	2138	0	
0	661	1875	0	106	0	1362	0	2007	430148	0	2000	0		
	0	1235		0		0		0	0		0			
		0												

Table A.70: Routes of solution D, 50_d2_tw4 ($Z2$ vs $Z5$).

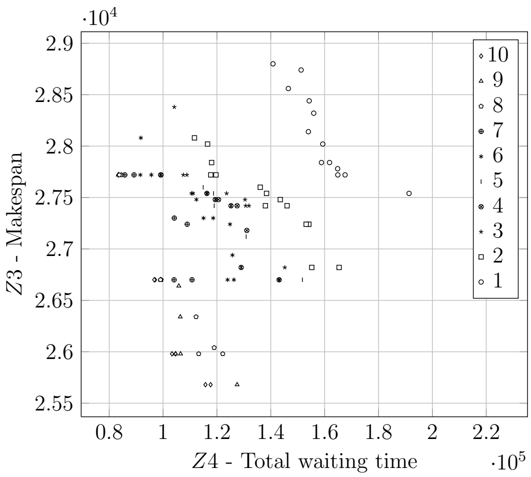

Figure A.141: Front progression of 50_d2_tw4 for $Z4$ vs $Z3$.

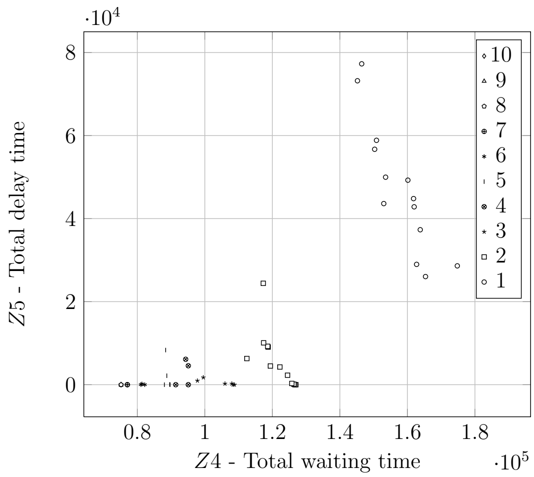

Figure A.142: Front progression of 50_d2_tw4 for $Z4$ vs $Z5$.

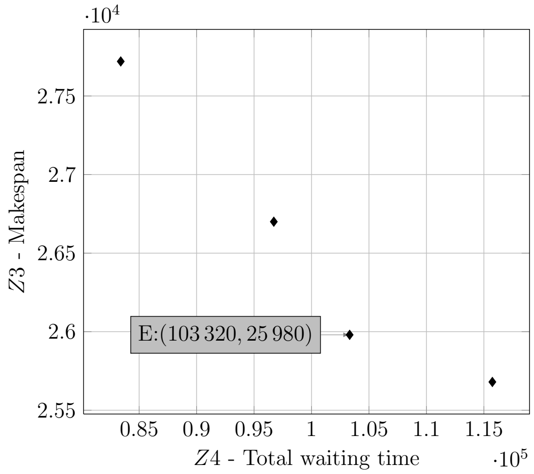

Figure A.143: Final approximation front of 50_d2_tw4 for $Z4$ vs $Z3$.

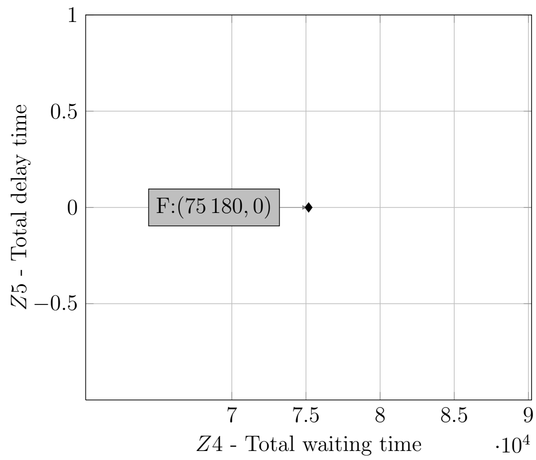

Figure A.144: Final approximation front of 50_d2_tw4 for $Z4$ vs $Z5$.

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**	**V13**	**V14**	**V15**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1725	1897	2044	1389	1703	1678	430030	1870	974	1173	1203	1777	1235	649	1384
1888	2121	948	486	106	430148	1588	430378	430471	2104	1509	1463	661	856	2138
430761	1686	482	669	1721	2107	1781	1362	2007	1714	430804	907	2149	1813	0
1875	0	0	0	2003	2073	0	0	0	0	0	0	430625	0	
2000				0	2152							0		
0					0									

Table A.71: Routes of solution E, 50_d2_tw4 ($Z4$ vs $Z3$).

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**	**V13**	**V14**	**V15**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1384	1389	430030	2107	1888	2000	2121	856	1203	1875	1703	2104	974	1173	1813
1777	1686	482	430148	1678	1362	106	1725	661	907	430471	1870	1588	1897	0
1463	2007	2003	1235	2044	430378	1781	649	669	2149	430804	2073	2138	1714	
0	0	0	1721	486	430761	2152	0	0	430625	0	0	0	0	
			948	0	1509	0			0					
			0		0									

Table A.72: Routes of solution F, 50_d2_tw4 ($Z4$ vs $Z5$).

A.13 250_d2_tw1

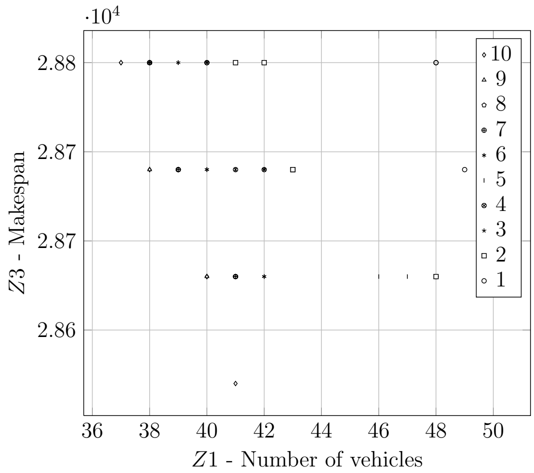

Figure A.145: Front progression of 250_d2_tw1 for $Z1$ vs $Z3$.

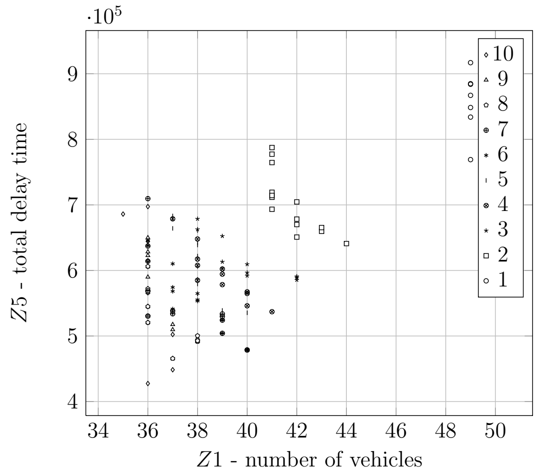

Figure A.146: Front progression of 250_d2_tw1 for $Z1$ vs $Z5$.

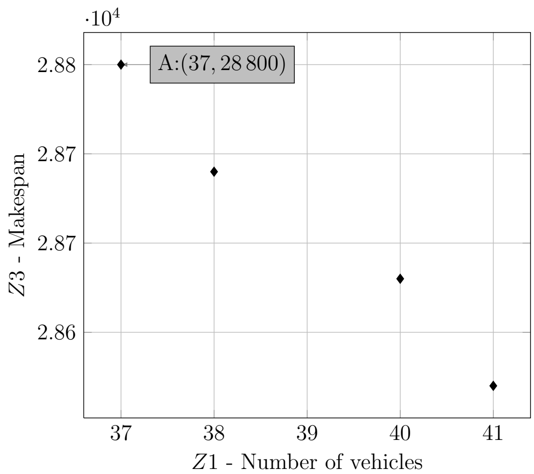

Figure A.147: Final approximation front of 250_d2_tw1 for $Z1$ vs $Z3$.

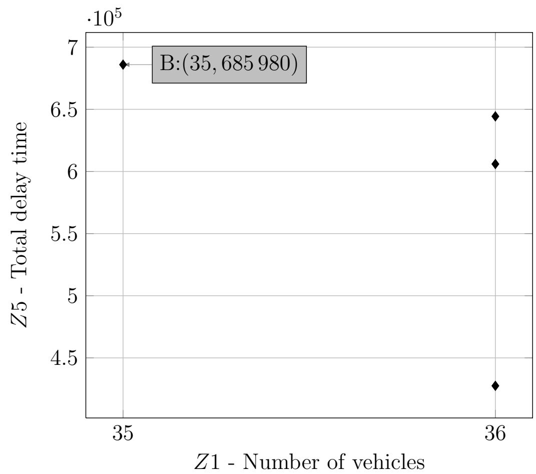

Figure A.148: Final approximation front of 250_d2_tw1 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13
0	0	0	0	0	0	0	0	0	0	0	0	0
430811	430108	1618	430793	1610	430471	2152	430618	924	430659	483	1630	1203
1769	2020	1938	948	1888	1509	430017	430030	1028	2091	1985	430378	1273
2104	2039	1613	1166	430148	1298	2136	1240	1714	579	430809	1528	1961
324	1140	192	661	2060	1707	1679	2048	491	1184	430760	1871	0
2073	1327	1201	941	1890	1897	1979	0	1729	1865	1463	2148	
1197	1102	430012	1450	2097	1813	1454		430221	0	0	1872	
968	0	0	2008	1808	1471	1877		1526			2134	
1448			1495	1524	669	430625		0			0	
581			649	1678	0	1508						
2122			1658	907		2000						
430465			0	1726		944						
0				0		0						

Table A.73: Routes of solution A – Part 1, 250_d2_tw1 ($Z1$ vs $Z3$).

V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25
0	0	0	0	0	0	0	0	0	0	0	0
1671	1461	2159	1777	1717	1219	1996	866	1803	306	1959	1116
1843	1007	1870	838	1917	1987	453	1384	1162	1601	1432	430529
430450	1464	2044	1761	1703	1937	1941	1582	2121	1686	1040	1743
1369	2100	974	0	430749	1474	1677	326	430804	562	1835	430353
1692	645	0		0	1749	1344	430089	1789	1776	1571	1665
1588	312				1619	639	430141	0	0	2007	2103
430389	1690				856	0	1866			106	695
2081	1642				2126		430761			1750	244
1900	236				0		0			1362	0
0	0									0	

Table A.74: Routes of solution A – Part 2, 250_d2_tw1 ($Z1$ vs $Z3$).

V26	V27	V28	V29	V30	V31	V32	V33	V34	V35	V36	V37
0	0	0	0	0	0	0	0	0	0	0	0
2084	1928	1948	1576	430792	1688	2003	1965	815	1348	2138	1635
1389	430049	430603	486	430806	1697	1584	250	1966	1781	785	1551
839	1621	1933	833	1559	2145	1094	430155	1602	430014	1873	1583
1173	2107	482	430346	0	535	2165	1757	1235	1721	2149	1458
553	2031	451	1879		1049	300	1029	765	322	0	749
985	593	1001	430532		1875	1867	1333	221	430958		0
0	133	315	0		0	1623	0	1194	0		
	1804	430626				1925		0			
	430278	0				0					
	1725										
	1130										
	1294										
	1482										
	0										

Table A.75: Routes of solution A – Part 3, 250_d2_tw1 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12
0	0	0	0	0	0	0	0	0	0	0	0
1184	1843	430529	315	221	1635	1613	430089	430806	1879	2003	1996
430659	430793	535	593	1865	1987	1743	430049	2031	430471	1298	430141
1623	1658	1757	1621	1551	1240	1203	1875	1389	2060	300	974
2152	1201	430017	451	453	2091	430030	1116	1688	2100	2073	2149
1941	1559	661	1197	2097	1777	0	1474	2138	1959	1333	2145
749	430012	430346	2122	430465	1937		1872	1671	1835	562	0
1677	0	430626	765	695	0		0	649	907	941	
1450		1463	1194	2148				2103	2020	1166	
0		491	430809	1776				833	1173	1726	
		0	1900	0				0	133	0	
			0						645		
									0		

Table A.76: Routes of solution B – Part 1, 250_d2_tw1 ($Z1$ vs $Z5$).

V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24
0	0	0	0	0	0	0	0	0	0	0	0
250	1001	2084	483	1866	579	1966	924	430014	2159	1471	1917
2165	1219	866	326	1583	1928	2007	430148	1948	1610	2121	1703
1162	1781	1890	1369	2081	1867	2039	2136	1933	322	2044	430749
2000	1870	430353	1344	1804	1235	486	430450	1721	430378	0	0
1524	192	1571	1454	1461	1761	1690	1094	2134	430532		
1679	430804	1750	1725	430155	669	106	1925	1327	430625		
1102	1871	1362	0	1877	1985	0	430389	1576	1464		
1665	0	1692		1029	244		1294	1495	0		
236		312		2008	0		1130	1642			
0		1007		0			1584	1749			
		0					0	0			

Table A.77: Routes of solution B – Part 2, 250_d2_tw1 ($Z1$ vs $Z5$).

V25	V26	V27	V28	V29	V30	V31	V32	V33	V34	V35
0	0	0	0	0	0	0	0	0	0	0
306	1888	1630	1348	1979	430618	553	1965	1803	430108	430761
815	1707	430811	1961	1482	1686	1140	430792	785	1938	2048
839	1602	2104	1049	1588	1601	1717	1873	838	1526	1697
1448	430958	324	1273	581	1714	1897	1509	2107	1384	430603
1678	430221	1813	0	968	0	0	0	0	1618	1458
1432	1028	430760		948					430278	482
856	985	1789		0					1769	0
1808	0	1528							1729	
2126		944							0	
1619		1508								
1582		639								
1040		0								
0										

Table A.78: Routes of solution B – Part 3, 250_d2_tw1 ($Z1$ vs $Z5$).

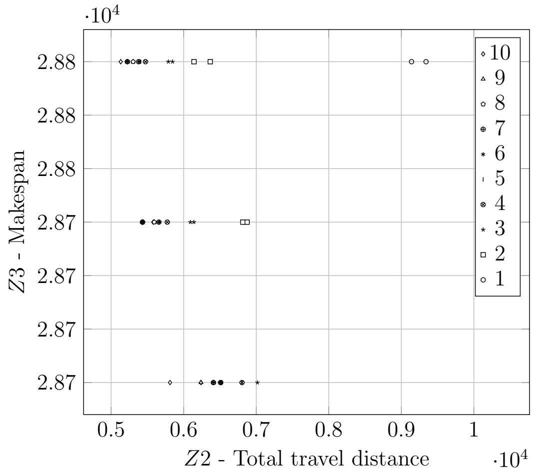

Figure A.149: Front progression of 250_d2_tw1 for $Z2$ vs $Z3$.

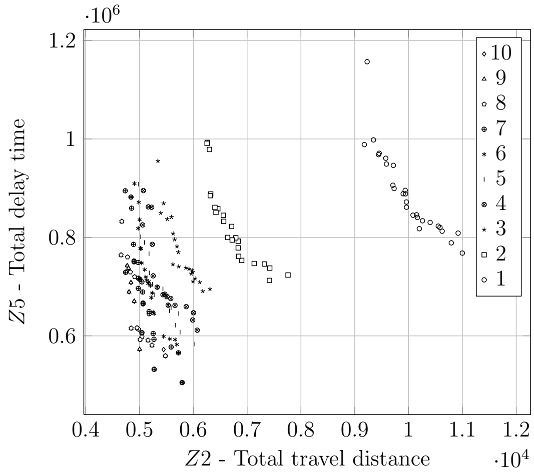

Figure A.150: Front progression of 250_d2_tw1 for $Z2$ vs $Z5$.

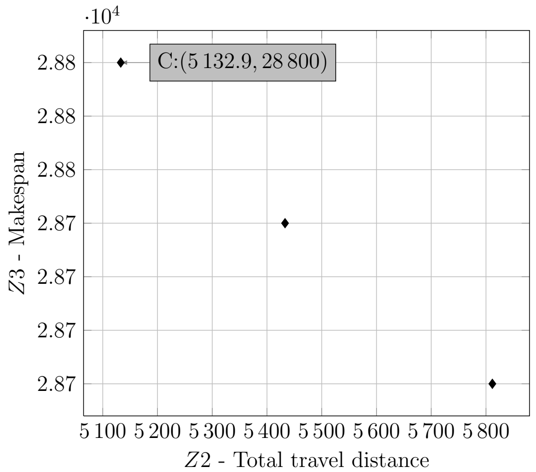

Figure A.151: Final approximation front of 250_d2_tw1 for $Z2$ vs $Z3$.

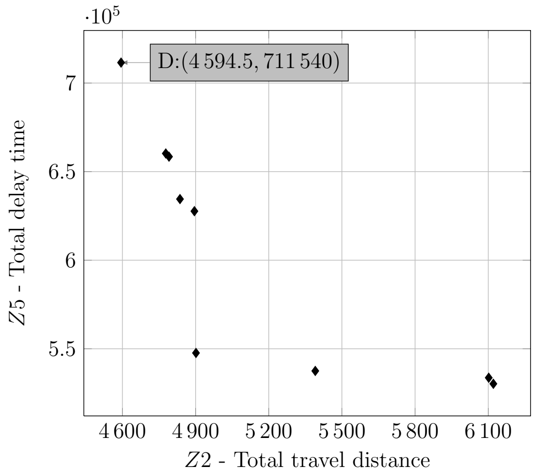

Figure A.152: Final approximation front of 250_d2_tw1 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1697	1630	322	2084	785	1959	1369	1996	838	483	430353	133	1803	839	1495
1235	430618	1789	985	579	1524	2122	430804	1781	815	1750	1458	1219	1389	2007
1875	430529	430958	1776	1987	2100	2008	1184	1985	2159	430603	1450	1743	1201	968
1870	1888	1872	430532	1928	1690	645	1613	2107	1965	1583	430465	1961	2121	1298
1551	649	1871	244	2031	1173	1464	430659	315	2138	1890	1040	1761	2044	1602
482	1007	0	1925	0	0	1692	0	221	1273	1528	106	1474	1804	430760
451	1948		1900			941		0	1703	1327	1102	2081	0	0
0	430811		0			0			250	1166	639	0		
	1658								0	2073	1665			
	0									2060	300			
										0	430450			
											0			

Table A.79: Routes of solution C – Part 1, 250_d2_tw1 ($Z2$ vs $Z3$).

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1635	1642	430378	553	430278	192	1240	1508	924	491	2103	1688	312	2091	1601
430089	2104	1966	1344	1028	765	1917	2000	430017	1432	1461	430761	1619	1777	1686
430749	1813	1094	1726	1714	669	1938	1162	430148	2126	1835	1867	2039	2048	430014
1621	1843	1294	1140	1879	1865	0	2134	535	1877	1671	430141	1482	1937	1757
0	324	1584	1808	236	0		1678	1941	0	0	430792	0	430049	1362
	0	0	0	0			453	948			1623		0	0
							2003	1679			0			
							974	944						
							593	1933						
							430806	0						
							1116							
							2148							
							1130							
							0							

Table A.80: Routes of solution C – Part 2, 250_d2_tw1 ($Z2$ vs $Z3$).

V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1717	430108	1588	1526	430471	1576	430030	2152	1559	430221	430155	1729	2149	1769
866	1203	2165	486	1509	833	306	562	1471	2136	1194	1725	1348	1721
430346	1049	1582	1454	1749	1873	326	1333	2097	1677	0	430793	1001	0
1571	2145	1029	907	1610	430809	0	581	0	0		1384	1618	
856	0	0	1707	695	430626		1197				0	1866	
661			1448	430389	430625		1979					430012	
1463			0	0	1897		0					2020	
0					0							749	
												0	

Table A.81: Routes of solution C – Part 3, 250_d2_tw1 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12
0	0	0	0	0	0	0	0	0	0	0	0
1781	430049	1686	1803	1584	1890	985	1551	1162	2149	1219	1865
1873	1928	1601	1240	312	1528	1333	1116	2134	315	2159	1866
2148	430108	430618	2003	1619	1588	1524	1870	1461	2107	579	1965
1804	815	2136	1996	2104	2165	2020	669	1717	1348	2091	1917
1028	1273	1450	1474	430760	1294	0	192	948	785	430659	1703
322	1961	0	0	244	1094		0	1750	2048	2044	430749
430221	1049			0	1130			2122	2145	1872	0
1714	0				2097			1678	2031	0	
491					0			1173	593		
639								2060	0		
0								1369			
								430465			
								1900			
								0			

Table A.82: Routes of solution D – Part 1, 250_d2_tw1 ($Z2$ vs $Z5$).

V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24
0	0	0	0	0	0	0	0	0	0	0	0
430793	944	1508	1576	430529	535	1966	1879	1933	430603	430378	1001
1495	2000	1610	833	924	1888	1679	2039	1007	695	1571	1688
866	1843	1677	968	430017	1757	1448	1432	1925	1482	1140	1184
1726	1454	1808	430450	430014	1362	907	562	430353	1690	1463	1618
300	1344	2007	649	430030	430148	1959	1102	553	1749	0	1867
1707	1665	430471	1029	1630	430804	2008	430346	1789	0		430155
1835	1602	430532	1464	430625	1559	2073	1871	0			2100
856	0	106	133	645	1471	1948	2081				1897
2103		1776	1197	0	221	324	0				1692
1384		0	1166		0	1813					1979
1298			0			0					941
1582											1040
0											0

Table A.83: Routes of solution D – Part 2, 250_d2_tw1 ($Z2$ vs $Z5$).

V25	V26	V27	V28	V29	V30	V31	V32	V33	V34	V35	V36
0	0	0	0	0	0	0	0	0	0	0	0
1583	2126	430806	306	581	1635	839	2152	1697	1769	1941	749
430278	1458	1621	483	1642	326	1875	2084	430141	453	1725	974
1729	236	451	1743	430811	1938	838	1671	765	1937	0	0
1526	430389	1201	1623	1509	2138	1613	1721	1194	1777		
430958	430626	2121	1389	0	250	430761	1877	0	1987		
0	0	430792	430809	0	1203	1985	486		430089		
		0	1761	0	1235	0	1327		0		
			482	0	430012	0	661				
			0	0	0	0	1658				
							0				

Table A.84: Routes of solution D – Part 3, 250_d2_tw1 ($Z2$ vs $Z5$).

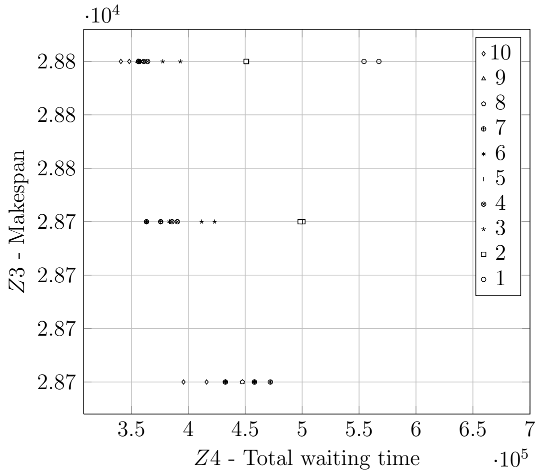

Figure A.153: Front progression of 250_d2_tw1 for $Z4$ vs $Z3$.

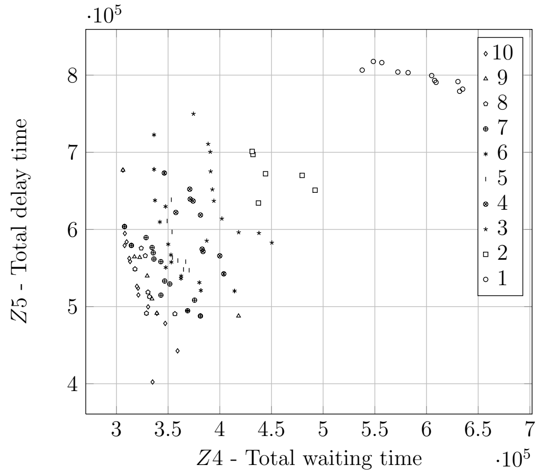

Figure A.154: Front progression of 250_d2_tw1 for $Z4$ vs $Z5$.

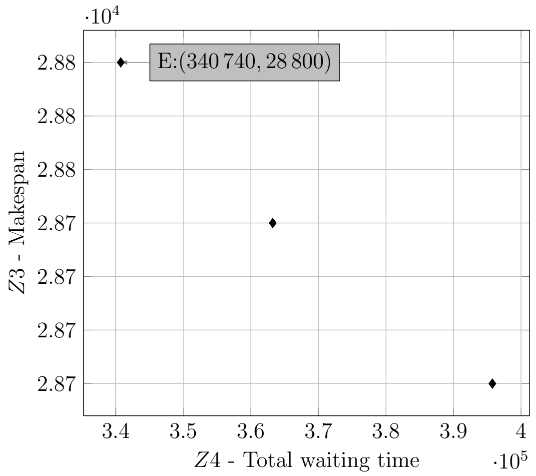

Figure A.155: Final approximation front of 250_d2_tw1 for $Z4$ vs $Z3$.

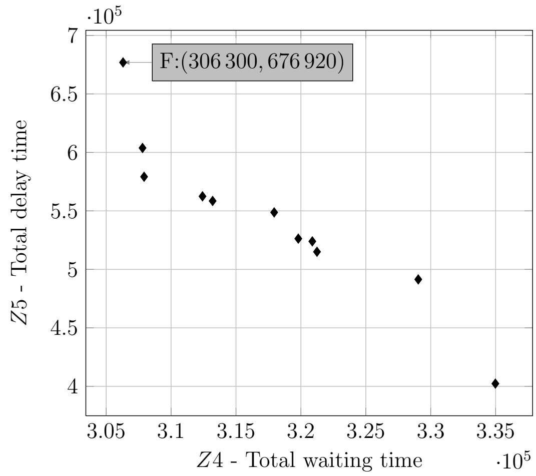

Figure A.156: Final approximation front of 250_d2_tw1 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13
0	0	0	0	0	0	0	0	0	0	0	0	0
1743	948	1461	1866	1703	1875	1219	250	1184	1803	815	1717	1781
1094	1610	1692	1769	1777	1116	2165	1697	1671	1843	1333	236	315
1961	1679	645	1583	2107	244	2100	1996	430353	1925	695	106	1001
1049	2134	1582	833	0	2148	430155	2044	785	430958	1482	941	430811
1273	2000	1509	430626		649	2081	1235	430792	1130	1028	1384	1559
0	1835	944	1584		1665	2097	1867	451	1327	0	1879	1619
	1588	866	1369		1979	553	1471	1872	1432		1458	2039
	430465	430221	430532		1524	0	0	0	0		581	0
	1749	1789	1040		0						1166	
	430793	2003	1776								1162	
	1726	1900	639								2008	
	0	0	1102								1933	
			1808								0	
			0									

Table A.85: Routes of solution E – Part 1, 250_d2_tw1 ($Z4$ vs $Z3$).

V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26
0	0	0	0	0	0	0	0	0	0	0	0	0
326	430108	1635	1623	839	1877	1613	2138	579	1618	2159	306	2152
430278	1348	430089	2145	968	1463	430030	1630	1721	1928	1621	1985	661
491	2091	1029	2060	1959	1707	1203	1890	1966	1576	1551	856	430603
430378	1938	1362	430760	1714	2103	0	2007	1678	1495	749	1897	486
0	430049	0	838	1690	0		430346	430017	453	1677	322	482
	2121		593	1888			430625	1686	1508	1173	1140	1464
	0		430806	430014			312	535	1602	1197	985	0
			1344	430529			0	0	430389	2122	1007	
			0	0					0	0	0	

Table A.86: Routes of solution E – Part 2, 250_d2_tw1 ($Z4$ vs $Z3$).

V27	V28	V29	V30	V31	V32	V33	V34	V35	V36	V37	V38
0	0	0	0	0	0	0	0	0	0	0	0
430761	430148	430618	2104	221	2136	1865	300	483	924	1526	192
1389	1948	1688	430749	1750	669	430659	1298	765	1987	2073	1528
1658	1813	1454	1240	1757	430141	1941	1965	430012	1873	1571	562
1201	133	1448	2048	1917	974	1601	2084	1294	1870	2126	0
2020	1194	907	1937	430809	0	1725	324	0	1761	1450	
1804	2149	430471	0	2031		0	1642		1729	0	
0	430804	430450		1871			0		0		
	1474	0		0							
	0										

Table A.87: Routes of solution E – Part 3, 250_d2_tw1 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13
0	0	0	0	0	0	0	0	0	0	0	0	0
2149	2138	1985	1781	1389	1937	2107	483	1623	430761	1865	1803	2003
326	1618	2165	1777	430806	2031	451	1996	2081	1866	974	430014	430353
453	1948	581	2048	1474	593	430793	1871	430603	968	192	1327	430141
2084	430030	1454	856	2148	430811	1925	1872	695	430618	1235	1298	430804
430378	306	948	1509	1714	1635	1602	1665	669	1630	1900	579	2073
322	1776	1707	0	562	1201	430958	1571	0	1240	0	430012	0
1804	0	1040		2008	2121	236	430278	0	785		430809	
0		106		1690	1028	1789	1094	0	1928		0	
		0		2122	244	0	0		0			
				1979	0							
				0								

Table A.88: Routes of solution F – Part 1, 250_d2_tw1 ($Z4$ vs $Z5$).

V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25
0	0	0	0	0	0	0	0	0	0	0	0
2159	1116	1938	1917	1688	1184	1987	430148	430049	535	1757	2152
1870	661	1448	1703	1875	924	1559	1933	2136	1613	1965	1194
1471	133	1528	430749	1344	2145	1508	221	1879	430017	1384	430346
765	2060	1582	0	1692	1273	2134	1621	1463	1961	1333	1140
0	430389	2100		1813	1203	944	1843	312	2091	1029	0
	430792	1526		553	1049	1750	2020	1464	1001	1482	
	839	2007		324	0	0	1877	1897	0	1588	
	1162	1524		0			1677	1458		0	
	941	907					866	1450			
	0	1808					0	0			
		430465									
		1362									
		0									

Table A.89: Routes of solution F – Part 2, 250_d2_tw1 ($Z4$ vs $Z5$).

V26	V27	V28	V29	V30	V31	V32	V33	V34	V35	V36	V37
0	0	0	0	0	0	0	0	0	0	0	0
1888	430471	430659	1743	250	815	1219	430108	1348	1873	1461	1619
430089	1495	430221	1867	1671	430626	1697	315	1941	1679	2104	0
2103	1959	1749	1576	491	2000	1584	639	1601	1432	1835	
1007	430450	1583	1369	1686	1551	1966	1717	1642	838	430760	
0	1197	1769	1173	430529	2097	1294	482	0	2044	1890	
	300	0	645	0	1725	1678	649		1761	2039	
	1729		2126		1130	1726	1610		0	486	
	1721		1166		0	430155	430532			985	
	430625		833			1658	0			749	
	1102		0			0				0	
	0										

Table A.90: Routes of solution F – Part 3, 250_d2_tw1 ($Z4$ vs $Z5$).

A.14 250_d2_tw2

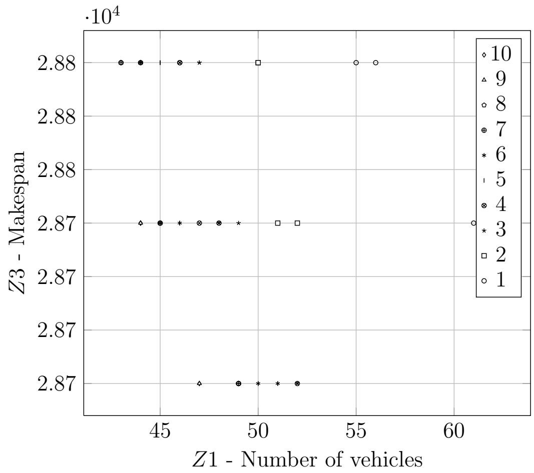

Figure A.157: Front progression of 250_d2_tw2 for $Z1$ vs $Z3$.

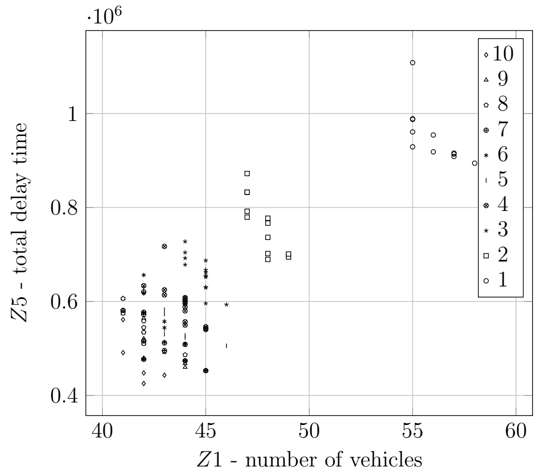

Figure A.158: Front progression of 250_d2_tw2 for $Z1$ vs $Z5$.

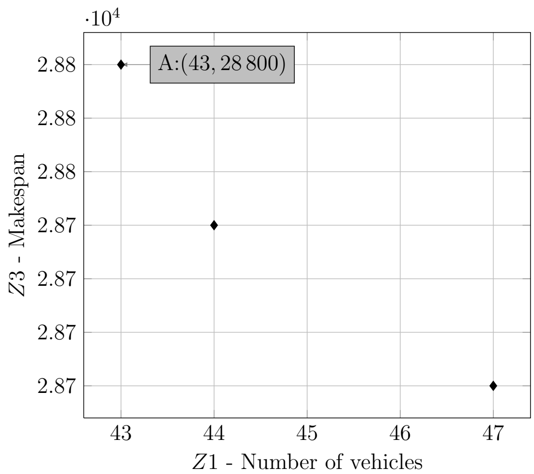

Figure A.159: Final approximation front of 250_d2_tw2 for $Z1$ vs $Z3$.

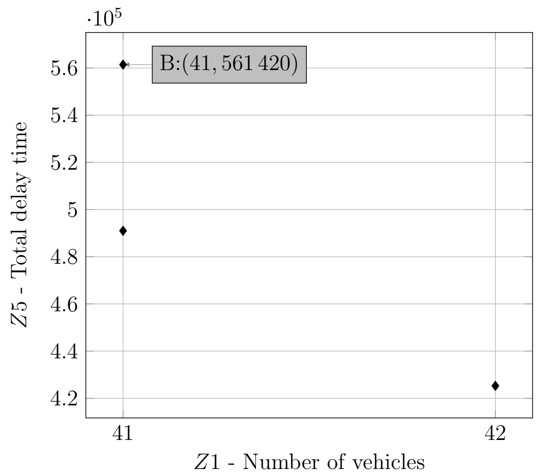

Figure A.160: Final approximation front of 250_d2_tw2 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2048	1781	430811	1692	1697	430108	1630	1610	1843	430618	430017	2152	1717	1618	326
1582	430806	430760	430793	1959	1001	1203	430378	1658	2136	1671	2000	1835	2134	453
1528	2104	244	1750	1979	1996	1273	1925	2103	1007	1897	1162	1454	1933	430049
1813	1384	236	2100	2122	1102	0	1872	1686	430625	1584	1677	1877	1327	562
1714	2008	1776	1619	430958	430346		1804	0	1576	1294	486	1464	430030	2020
106	1948	1333	1725	430221	1726		0		907	482	1679	1461	306	1448
1040	2007	1369	1130	1729	1808				430465	1621	968	1362	0	661
1721	1166	2126	1094	0	2039				553	430471	1678	0		0
2060	0	645	0		0				430353	0	0			
1571		0							866					
833									0					
1432														
0														

Table A.91: Routes of solution A – Part 1, 250_d2_tw2 ($Z1$ vs $Z3$).

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1526	315	1703	430529	430014	2084	815	2149	1688	1875	1928	2003	221	1551
1495	2031	430809	430450	1941	1029	1873	1870	2107	1938	1865	856	1463	430749
1588	579	0	1197	1601	948	430141	1194	1867	1201	1985	1524	1690	1917
1482	483		300	944	430278	1559	0	1184	974	1509	1642	2097	0
2081	1937		639	0	0	1866		1583	593	250	1474	0	
430389	1116		581			0		430626	0	1743	0		
0	785		749					1240		2159			
	430792		1749					430012		2091			
	0		1871					0		430089			
			0							0			

Table A.92: Routes of solution A – Part 2, 250_d2_tw2 ($Z1$ vs $Z3$).

V30	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1635	1890	322	1613	839	1965	924	2138	535	1803	1987	430148	1348	1235
2073	430155	491	430659	430804	1344	1508	2145	1219	2044	1140	1389	1966	1049
1707	133	1789	192	669	1173	1888	1879	765	312	1665	1777	695	1961
1769	1450	0	2121	430761	324	1757	430532	451	0	985	1471	1761	0
1900	649		1623	0	0	1028	1602	2165		0	0	0	
0	0		0			838	430603	0					
						2148	1458						
						0	941						
							1298						
							0						

Table A.93: Routes of solution A – Part 3, 250_d2_tw2 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1688	1867	2149	221	1965	430529	1813	1996	1623	1835	1184	2048	2003	250
430532	1804	1873	866	579	1875	324	1007	1116	192	2165	1551	1870	1630
1040	244	430155	1166	785	593	2104	430626	1933	2044	2081	430141	1559	430625
1776	236	1029	300	430792	430806	1471	1925	2134	2031	1890	430089	0	1671
1948	639	985	907	0	1985	765	430278	2000	0	1130	0		430030
1726	1524	0	1464		2136	1866	1526	430760		1725			2100
1808	312		1917		2152	1865	0	661		1463			0
1959	1665		430749		1173	0		1102		1528			
1678	0		0		2103			1642		2122			
1495					1384			1789		0			
430014					0			0					
1333													
695													
0													

Table A.94: Routes of solution B – Part 1, 250_d2_tw2 ($Z1$ vs $Z5$).

V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1621	2138	924	1610	1635	1001	1721	2126	306	1508	430148	1389	1219	430108
451	1703	669	1201	839	315	1576	430346	1458	430471	1362	1928	430450	430017
1602	2091	1690	326	430659	2107	968	486	1582	1781	944	2060	430793	535
1584	0	0	0	1235	833	1769	0	430389	1613	1273	453	1482	645
133				430809	1298	1583		430603	322	1203	1743	2097	1750
1966				1474	948	1344		482	553	0	430049	430958	1749
1294				1872	2020	1571		1697	430811		0	0	1677
1327				2148	749	2007		0	0				562
581				1871	106	0							0
1140				0	1461								
1197					0								
856													
0													

Table A.95: Routes of solution B – Part 2, 250_d2_tw2 ($Z1$ vs $Z5$).

V29	V30	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41
0	0	0	0	0	0	0	0	0	0	0	0	0
2145	1348	941	2084	1937	1897	1979	1094	1618	430353	1757	1803	1888
483	1777	1941	1049	1240	1714	1679	838	1450	1369	1454	491	1843
430618	1194	2008	1961	430804	1879	1162	2121	1692	2073	1619	430221	815
1601	0	1707	0	0	1509	1938	430465	430012	1432	1729	1028	2159
1686		0			430378	1987	0	1877	1717	0	0	430761
2039					649	1761		0	0			974
0					1900	1448						1588
					0	1658						0
						0						

Table A.96: Routes of solution B – Part 3, 250_d2_tw2 ($Z1$ vs $Z5$).

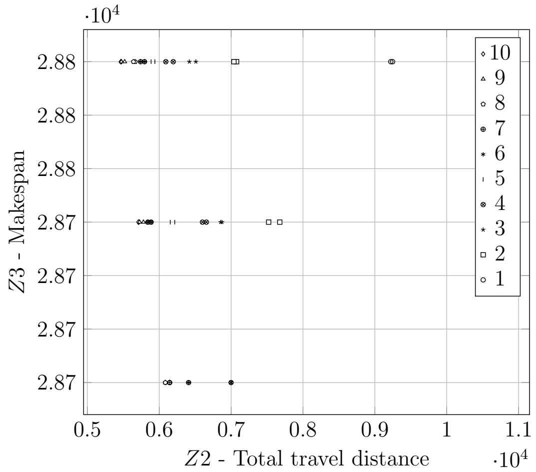

Figure A.161: Front progression of 250_d2_tw2 for $Z2$ vs $Z3$.

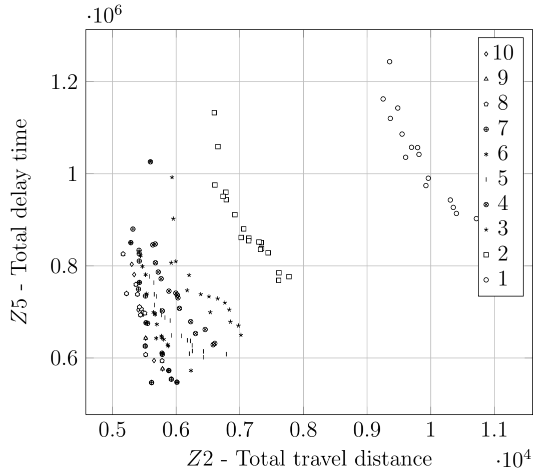

Figure A.162: Front progression of 250_d2_tw2 for $Z2$ vs $Z5$.

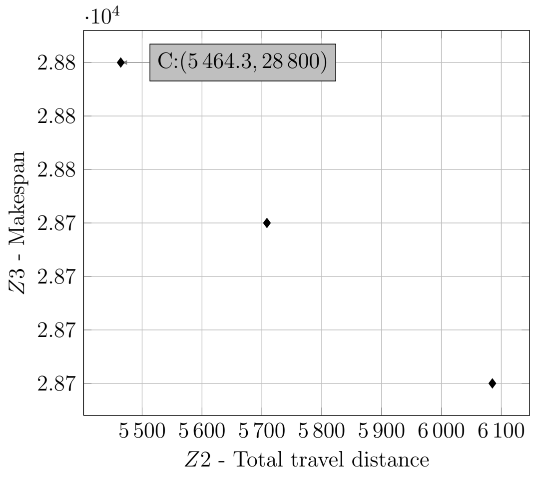

Figure A.163: Final approximation front of 250_d2_tw2 for $Z2$ vs $Z3$.

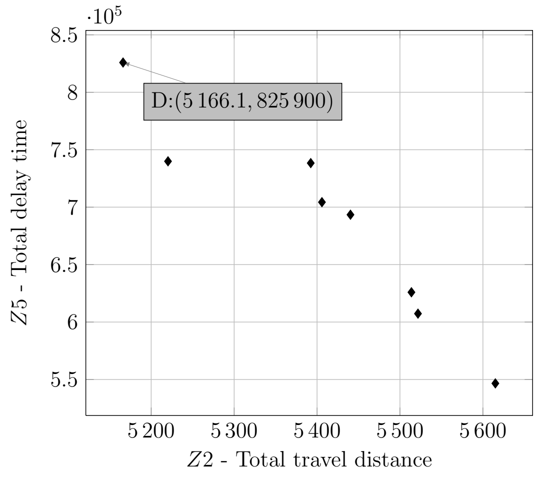

Figure A.164: Final approximation front of 250_d2_tw2 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1235	2149	430017	430378	430618	1094	430049	1917	1635	765	1769	1870	1642	1610	1551	430278
593	2048	1630	1879	1601	1925	430108	1703	326	2107	866	482	430760	1843	1348	1482
1474	1203	1362	312	1941	1102	1743	0	815	221	1749	451	1690	1602	1688	1877
1725	1961	1933	941	645	1140	483		430749	315	1528	0	106	1194	430761	0
430221	0	2134	0	1450	0	1049		1618	2031	236		0	1781	669	
639		2000		0		1777		1900	0	856			1867	1471	
430532		833				0		0		0			1001	1116	
0		968											0	0	
		1679													
		430450													
		661													
		2122													
		1979													
		1007													
		1582													
		0													

Table A.97: Routes of solution C – Part 1, 250_d2_tw2 ($Z2$ vs $Z3$).

V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30	V31	V32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1966	322	1448	985	974	1029	553	1389	1344	430346	430030	1959	430659	453	1803	1571
1872	1028	1576	1463	2044	1677	1813	1987	2020	1495	430471	2136	1623	2138	1240	948
430626	1714	1166	2097	1184	1524	430958	2145	1454	1808	0	2152	1873	1965	430141	430603
430353	1776	1197	0	0	0	430389	2121	0	1721		1757	1866	250	0	244
324	1040	1726				0	1761		0		1835	1865	2159		0
2081	0	1432					1621				1327	192	838		
2165		133					0				2084	2003	1201		
1583		430465									562	1996	1985		
1890		2060									1658	0	0		
1173		430793									1678				
1692		1871									0				
0		2148													
		1897													
		1464													
		1948													
		0													

Table A.98: Routes of solution C – Part 2, 250_d2_tw2 ($Z2$ vs $Z3$).

V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45	V46	V47
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430809	1508	1273	2039	924	1697	491	2073	1369	1384	1526	839	1750	306	1686
430012	1162	2091	430811	535	579	1804	2007	695	1294	430155	430792	2104	785	0
1613	1619	0	1298	430148	1928	1729	486	1707	2126	1717	1937	1461	430089	
0	1509		2100	430014	1938	1789	0	0	1333	430625	430806	2103	0	
	1584		907	430529	1219	0			2008	0	0	749		
	1130		1458	1888	1875				0			0		
	0		1588	1671	430804									
			944	581	1559									
			649	1665	0									
			0	300										
				0										

Table A.99: Routes of solution C – Part 3, 250_d2_tw2 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1757	2134	430760	1495	430030	1867	1948	430017	924	1116	1692	1219	326	430049	968
1658	2126	2104	856	2136	315	1458	1941	1630	1389	1872	430108	1559	1875	1678
2097	2000	1642	300	0	430659	1679	1671	1362	839	1130	785	430809	2149	1665
244	941	1789	1726		1201	1298	1582	1750	1987	430278	430792	1184	1688	1879
1610	430811	1463	1197		430089	1369	430465	2007	1928	1966	0	0	1001	1776
1508	0	0	1808		0	581	0	1454	1865	1094			491	639
0			1769			1432		430603	1348	1294			1028	1166
			1602			1528		0	1761	2148			1714	0
			1843			0			1621	0			1583	
			907						2100				0	
			486						645					
			0						0					

Table A.100: Routes of solution D – Part 1, 250_d2_tw2 ($Z2$ vs $Z5$).

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430450	1743	1803	553	1938	430148	306	430346	430529	322	2152	1813	1835	1866	833
1979	2138	483	1749	1937	430014	250	1897	1888	1344	1870	1584	1959	1623	866
1509	2159	1917	695	579	535	815	1464	133	948	192	1890	430155	2145	1524
2039	1240	430749	0	1697	1576	1273	1007	1327	0	1474	430389	1461	1235	1029
1571	2091	0		765	1877	1961	106	661		0	430626	2103	593	2084
430353	2048			221	1173	0	1040	749			1900	1677	430806	0
1526	0			430012	562	0	0	2020			430471	0	1551	
0				2031	0			1140			430532		451	
				0				2122			430958		2044	
								0			1482		2121	
											0		2107	
													0	

Table A.101: Routes of solution D – Part 2, 250_d2_tw2 ($Z2$ vs $Z5$).

V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430625	1933	430761	430618	453	1162	1448	430378	1729	1384	1781	2060	1588	312	974
2008	944	1613	1686	1471	1690	1450	1725	430221	1102	1965	1049	430793	0	430804
649	2003	1873	1601	1194	236	1333	1804	1925	2073	1703	1203	1721		0
324	1996	1618	1707	0	2165	985	0	0	1619	1777	0	0		
0	1635	1985	0		2081	1717			0	0				
	838	669			0	0								
	482	430141												
	1871	0												
	0													

Table A.102: Routes of solution D – Part 3, 250_d2_tw2 ($Z2$ vs $Z5$).

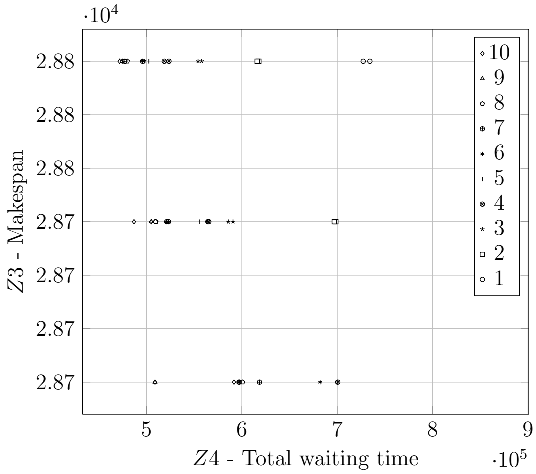

Figure A.165: Front progression of 250_d2_tw2 for $Z4$ vs $Z3$.

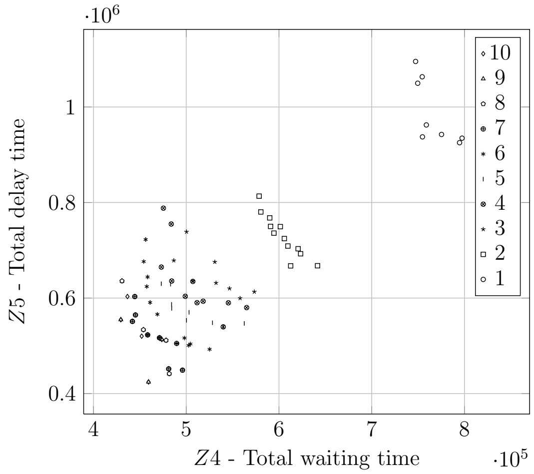

Figure A.166: Front progression of 250_d2_tw2 for $Z4$ vs $Z5$.

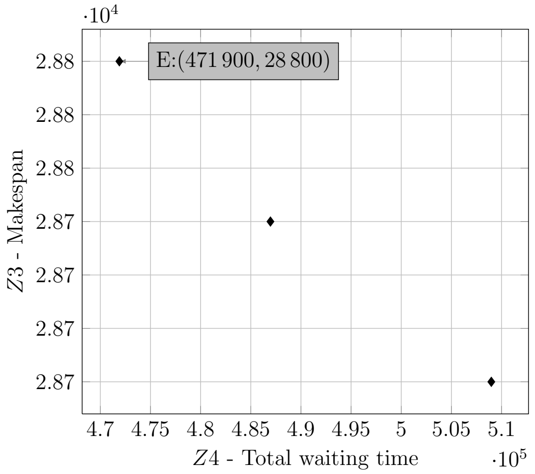

Figure A.167: Final approximation front of 250_d2_tw2 for $Z4$ vs $Z3$.

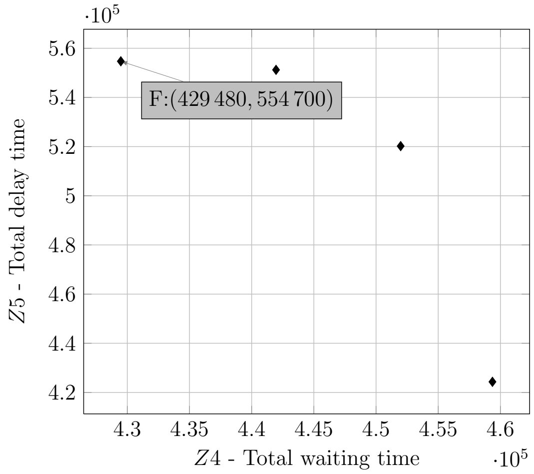

Figure A.168: Final approximation front of 250_d2_tw2 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1130	1875	815	1948	2165	2084	453	1959	2145	1389	1803	1781	1551	1866	2107
1871	315	430760	2091	1528	856	669	1835	430030	430806	430148	1454	430089	1584	1298
1804	322	2003	1240	1679	661	430012	1717	2097	1621	1888	1665	1559	1813	482
236	1166	1789	1937	553	1450	1867	1601	0	2031	974	1897	0	430811	451
1197	1294	948	0	1900	1464	0	1690		1917	1761	430465		1688	430389
968	0	1102		1602	1495		1776		1703	2081	0		1273	430603
833		0		1890	430346		0		0	0			1961	0
1576				0	2008								0	
2122					1333									
1979					0									
1721														
486														
1726														
1692														
649														
0														

Table A.103: Routes of solution E – Part 1, 250_d2_tw2 ($Z4$ vs $Z3$).

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1623	2048	2159	430049	430014	326	250	1965	1348	430017	579	2152	483	1873	1987
907	300	1877	1184	1327	1474	2039	324	839	765	430659	1743	2000	430761	1201
2136	866	1729	430618	430532	1162	1618	1028	430353	430804	562	430141	1658	2020	491
1750	1870	430626	430792	639	430471	785	1588	1642	1116	2073	1610	133	1933	0
1677	1471	0	430809	1686	192	2044	1524	1369	0	1344	1362	312	1094	
2148	0		593	1582	430278	0	1448	430958		0	0	2007	1872	
1571			0	1843	0		1432	0				985	0	
0				0			0					0		

Table A.104: Routes of solution E – Part 2, 250_d2_tw2 ($Z4$ vs $Z3$).

V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1938	924	1630	430108	1671	306	1865	430529	1219	1928	1635	944	430793	1194
1757	1941	1925	2149	1509	1966	430625	2134	2060	2104	1996	1808	838	2103
1526	430155	1725	1613	1001	430450	1777	1697	1714	1029	430749	1140	1583	0
1879	645	1482	1678	1235	2100	2121	2138	430221	1007	221	1461	0	
1619	0	0	244	1985	1049	0	1749	1707	695	0	1173		
1463			106	535	1203		1458	749	0		0		
581			1040	1508	0		941	0					
0			1384	0			1769						
			2126				0						
			430378										
			0										

Table A.105: Routes of solution E – Part 3, 250_d2_tw2 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1721	430761	2000	326	483	785	430806	1184	1635	430148	1875	1461	1362	430529
1610	535	639	1618	453	2044	1933	1094	1551	856	1697	2020	300	2104
1348	1602	106	315	1273	0	1116	1162	430278	1714	1688	1344	1959	430760
430792	1194	1776	430809	1203		833	430346	430958	1890	221	1474	1879	1948
974	1866	1040	581	0		1583	1140	1789	1813	941	1900	661	1450
2031	1298	2081	0			2097	312	0	765	866	0	2039	1464
0	0	1872				1941	0		482	430141		1509	1029
		1871				0			451	0		486	1678
		1130							0			1877	1726
		430450										944	968
		1679										2134	2122
		2003										0	430465
		430089											0
		2091											
		0											

Table A.106: Routes of solution F – Part 1, 250_d2_tw2 ($Z4$ vs $Z5$).

V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1865	1867	2159	250	1937	1613	1928	1888	1743	1803	2149	1001	430049	430618
2138	1294	1965	2107	1448	1870	1197	1808	1961	1938	2165	1621	1987	1769
430625	1482	430155	1623	430626	1584	1757	1843	2121	1508	1576	1985	1703	749
645	1665	1677	579	1327	2148	1630	236	0	1458	669	306	430804	1384
2007	1749	562	1559	2152	0	1369	1917		1750	593	1781	0	2100
491	0	0	553	2073		1524	430749		2008	1725	133		1166
430221			0	1454		1173	0		948	0	1007		0
0				649		0			1571		1582		
				0					1432		430793		
									0		1729		
											1925		
											244		
											0		

Table A.107: Routes of solution F – Part 2, 250_d2_tw2 ($Z4$ vs $Z5$).

V29	V30	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42
0	0	0	0	0	0	0	0	0	0	0	0	0	0
2048	1671	2145	430108	1219	1873	815	430014	2136	1707	1495	1240	1389	839
1528	924	1049	1897	1686	1526	1966	1835	430030	1235	1996	1201	430603	2084
1979	1777	1761	1690	1588	1333	322	907	430017	1601	985	0	430659	1471
430353	192	0	695	0	430532	1804	324	0	2103	430471		1619	1463
2060	0		430811		1692	1028	1717		1658	430378		1102	0
838			0		2126	430389	430012		0	0		0	
1642					0	0	0						
0													

Table A.108: Routes of solution F – Part 3, 250_d2_tw2 ($Z4$ vs $Z5$).

A.15 250_d2_tw3

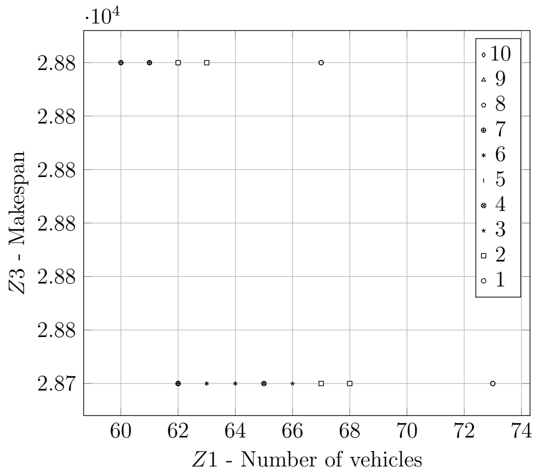

Figure A.169: Front progression of 250_d2_tw3 for $Z1$ vs $Z3$.

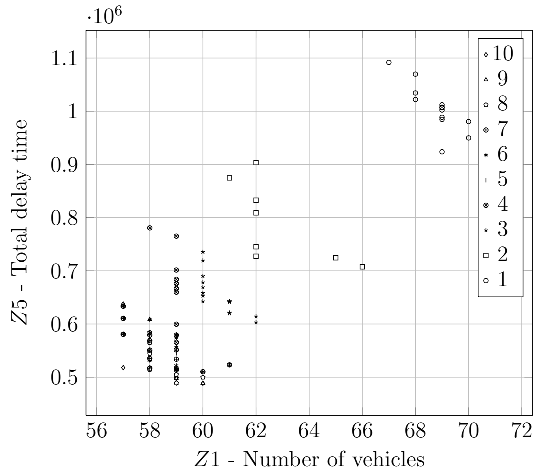

Figure A.170: Front progression of 250_d2_tw3 for $Z1$ vs $Z5$.

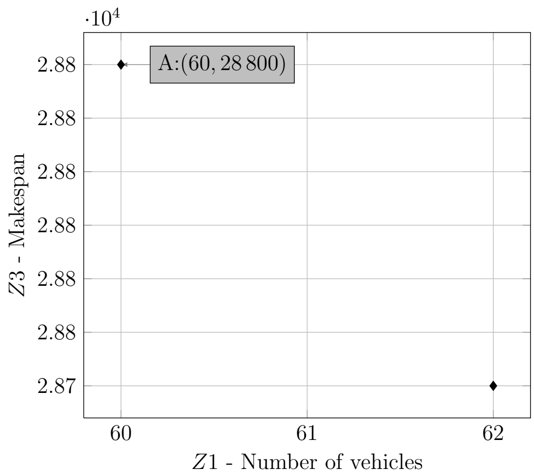

Figure A.171: Final approximation front of 250_d2_tw3 for $Z1$ vs $Z3$.

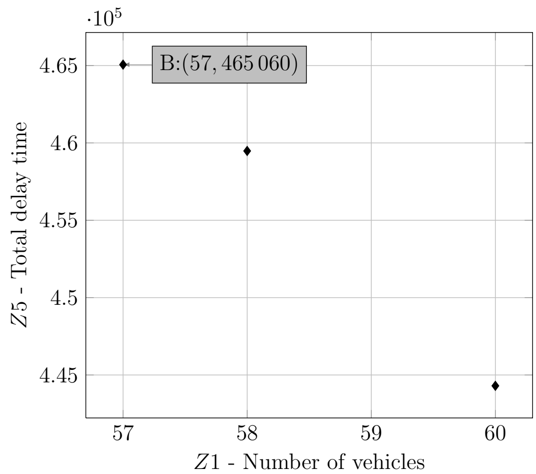

Figure A.172: Final approximation front of 250_d2_tw3 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430806	1948	924	430278	430811	579	1028	192	2060	907	430749	1384	1707	1551	430155
300	1582	815	749	453	1471	430030	1001	430618	430532	0	1559	1804	1173	1464
1495	430353	1049	0	1987	2148	0	1900	1686	1933		315	1201	941	1040
1642	2126	0		1777	1871		0	430017	669		0	430761	430346	2122
1665	1029			430049	0			0	1116			0	0	0
639	1362			0					0					
0	0													

Table A.109: Routes of solution A – Part 1, 250_d2_tw3 ($Z1$ vs $Z3$).

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
324	1658	2159	430089	1235	1508	661	430148	1917	2048	1985	1938	483	306	482
1327	1630	1965	451	430792	430471	430389	1888	0	1749	944	430625	985	833	430958
1389	856	785	0	1623	430760	1528	1461		1463	1166	1448	1007	106	0
1273	1870	0		0	1843	430626	2073		430465	1873	765	1619	1813	
0	1621				2165	1757	645		0	1743	1618	0	1875	
	0				1130	430659	2007			1721	0		2000	
					649	974	0			553			2121	
					312	0				1482			1474	
					1450					0			0	
					0									

Table A.110: Routes of solution A – Part 2, 250_d2_tw3 ($Z1$ vs $Z3$).

V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1996	250	838	326	430603	2138	1219	1526	491	968	535	2031	1583	1769	1671
1703	1697	244	2152	839	1203	1717	2008	1729	1197	948	2091	1966	1458	1094
0	1961	0	1333	486	0	2100	1571	1872	1524	1688	1937	1959	1369	1714
	0		1692	695		1761	0	0	1140	1294	0	1162	430378	1677
			430450	0		2097			1808	1602		1941	0	1690
			1726			0			0	322		1454		0
			0							430221		0		
										0				

Table A.111: Routes of solution A – Part 3, 250_d2_tw3 ($Z1$ vs $Z3$).

V46	V47	V48	V49	V50	V51	V52	V53	V54	V55	V56	V57	V58	V59	V60
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1928	1298	1890	1635	430529	2134	1679	2084	1897	1576	1601	1803	1613	1240	1877
2044	1678	2145	430793	1750	1584	1348	1879	2104	1835	430014	1725	133	0	2136
221	1776	430012	581	2039	2107	2103	1194	866	1344	0	1610	1866		0
0	2003	1781	1925	1432	430108	1102	1184	430804	2020		0	1867		
	430141	0	0	1979	562	0	0	2149	0			430809		
	593			1865	1509			0				0		
	0			1588	2081									
				1789	0									
				236										
				0										

Table A.112: Routes of solution A – Part 4, 250_d2_tw3 ($Z1$ vs $Z3$).

V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	**V11**	**V12**	**V13**	**V14**	**V15**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1803	430529	1630	430108	944	1686	1996	1219	1865	1688	1509	483	968	306	2081
1551	1362	1917	430049	451	2152	1583	1526	1040	300	430659	1870	1576	322	661
785	430760	0	907	1872	0	430792	1203	856	1588	1743	244	1610	430378	2003
1900	1471		669	1130		1623	0	1966	430221	1987	0	2165	0	2136
0	1613		1116	2060		0		1584	0	839		2044		1690
	0		0	430793				2073		838		1928		562
				1049				1789		430806		0		0
				0				0		0				

Table A.113: Routes of solution B – Part 1, 250_d2_tw3 ($Z1$ vs $Z5$).

V16	**V17**	**V18**	**V19**	**V20**	**V21**	**V22**	**V23**	**V24**	**V25**	**V26**	**V27**	**V28**	**V29**
0	0	0	0	0	0	0	0	0	0	0	0	0	0
430353	430017	1707	106	815	1461	1897	2091	1888	315	1933	1294	2048	948
1890	1781	1961	1776	430809	1658	2084	1937	1813	430532	1197	1327	430089	1298
2100	430958	0	1678	1618	1777	1769	0	2138	866	430603	649	1184	1454
2134	1173		833	0	1938	1102		553	133	312	1665	0	430346
1843	941		1166		0	1029		1619	1369	1464	639		430465
2039	0		1571			0		0	695	985	0		0
1642			2103						0	0			
1482			0										
0													

Table A.114: Routes of solution B – Part 2, 250_d2_tw3 ($Z1$ vs $Z5$).

V30	**V31**	**V32**	**V33**	**V34**	**V35**	**V36**	**V37**	**V38**	**V39**	**V40**	**V41**	**V42**	**V43**
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1697	1714	1965	430761	535	1635	430389	1985	2104	1621	2000	221	1001	974
453	1028	1671	765	645	1703	2148	430749	430030	1508	2159	2107	2031	2149
1094	2008	1559	1348	2122	0	1871	0	0	579	2007	1692	1721	0
1458	1463	1602	0	1432		236			2121	1524	1384	1879	
1140	0	0		1750		1528			1875	1979	1948	581	
1729				430811		2126			0	0	430012	1925	
0				430804		749					1235	0	
				2097		1726					0		
				0		0							

Table A.115: Routes of solution B – Part 3, 250_d2_tw3 ($Z1$ vs $Z5$).

V44	**V45**	**V46**	**V47**	**V48**	**V49**	**V50**	**V51**	**V52**	**V53**	**V54**	**V55**	**V56**	**V57**
0	0	0	0	0	0	0	0	0	0	0	0	0	0
250	1194	1808	1474	1804	430014	1835	1240	430148	1162	430471	1757	1959	192
430450	1867	430141	430625	1448	430626	1495	593	1389	1007	924	2020	1717	0
1725	0	1873	491	1582	430618	1601	0	2145	324	482	1333	1866	
486		0	1344	1679	1201	1941		326	1877	1761	0	1749	
0			0	1677	0	0		1273	430278	0		0	
				430155				0	0				
				1450									
				0									

Table A.116: Routes of solution B – Part 4, 250_d2_tw3 ($Z1$ vs $Z5$).

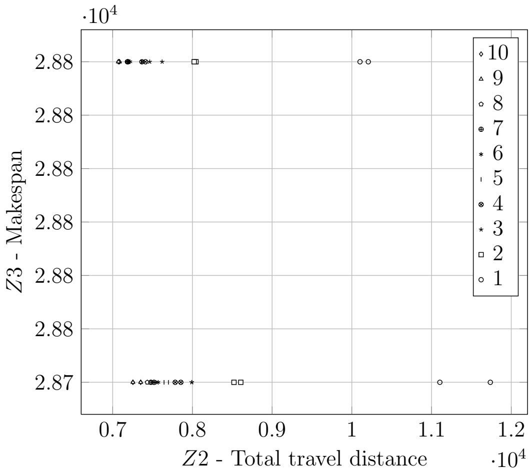

Figure A.173: Front progression of 250_d2_tw3 for $Z2$ vs $Z3$.

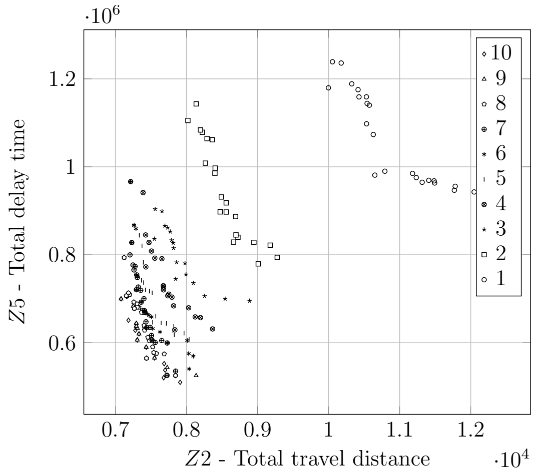

Figure A.174: Front progression of 250_d2_tw3 for $Z2$ vs $Z5$.

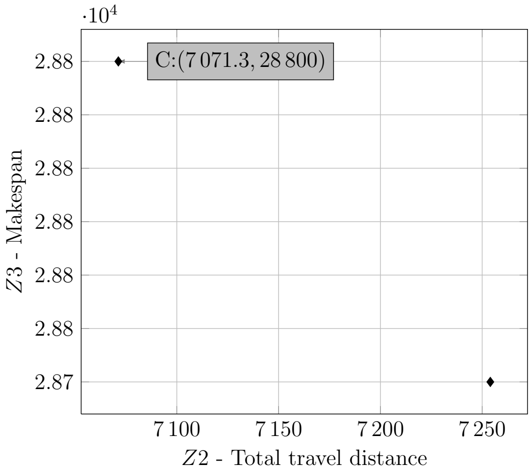

Figure A.175: Final approximation front of 250_d2_tw3 for $Z2$ vs $Z3$.

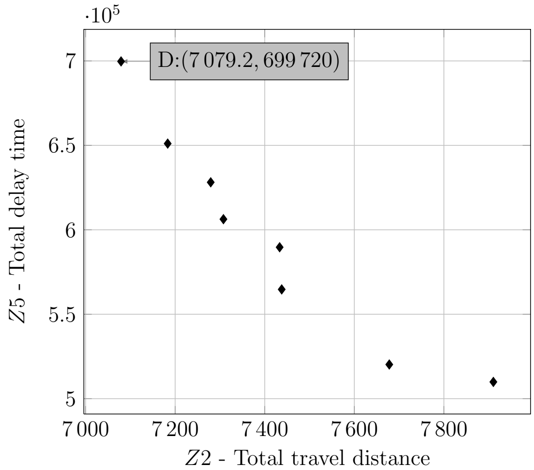

Figure A.176: Final approximation front of 250_d2_tw3 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1933	1966	300	491	1463	430014	1870	1690	1769	2138	2152	1865	1559	1725	1686	430792	430760
1610	1584	1197	430378	1482	924	451	1717	1344	250	985	1866	1348	1369	430148	1623	866
486	192	1897	0	1900	1703	0	833	430958	1219	1524	1389	0	0	535	0	1601
695	1116	1692		0	0		0	244	430749	0	1201			0		1888
0	0	1749						0	0		1621					1757
		1528									0					0
		1890														
		0														

Table A.117: Routes of solution C – Part 1, 250_d2_tw3 ($Z2$ vs $Z3$).

V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30	V31	V32	V33	V34
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430353	430809	1613	2136	430471	1996	1987	2008	1697	1873	1630	312	1729	2103	1619	326	1298
2104	2149	1471	2126	639	2091	1928	1642	1875	430659	1803	1384	322	581	2039	1203	948
324	0	221	1948	430465	430049	1635	0	430804	1777	483	1582	0	1726	1432	0	968
2020		1872	1658	1450	0	785		1618	1938	2121	0		0	1678		1102
1509		0	0	1454		0		0	0	1235				0		1040
430389				1173						0						236
0				1464												0
				0												

Table A.118: Routes of solution C – Part 2, 250_d2_tw3 ($Z2$ vs $Z3$).

V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45	V46	V47	V48	V49	V50
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1843	430012	430761	1665	1750	1714	1551	2073	430450	2044	2084	1495	562	453	856	1094
430793	1184	1194	1571	2134	1804	839	2007	907	2031	1327	430346	1677	579	661	974
1166	0	593	0	1508	1871	838	0	1679	1474	1761	1448	1362	2159	1029	1001
1979		0		1162	430626	2145		749	0	482	1813	0	1965	649	0
941				2000	430155	0		2060		0	0		1743	0	
645				1835	1461			0					430108		
430625				1333	1294								1917		
2122				133	0								0		
1721				1808											
430811				106											
2165				0											
2081															
0															

Table A.119: Routes of solution C – Part 3, 250_d2_tw3 ($Z2$ vs $Z3$).

V51	V52	V53	V54	V55	V56	V57	V58	V59	V60	V61	V62	V63	V64	V65	V66
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1671	306	430603	430089	1588	1879	815	2003	1526	430618	430030	1576	1583	430529	2048	1273
430017	1049	430278	1937	1925	1959	1961	765	430221	1941	0	1028	2097	430141	1240	0
1877	0	1789	0	2148	1776	0	315	1458	944		1781	2100	1985	0	
1140		0		0	553		2107	0	1007		1688	1707	1867		
430532					1602		0		0		669	0	0		
0					1130						430806				
					0						0				

Table A.120: Routes of solution C – Part 4, 250_d2_tw3 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1757	430529	1461	1495	1583	1776	430603	1619	430148	2020	1867	2121	430278	1344	1635
1671	2138	430346	1458	1769	1040	1679	430221	430030	430532	1985	430659	1130	430465	430089
924	483	695	866	1528	430471	1509	0	0	106	430012	0	1294	1665	1474
1686	1049	0	649	430811	1610	1576			1197	430626		1584	0	0
430017	0		2007	907	1140	1835			0	0		0		
0			0	2100	2103	133								
				430353	1362	948								
				1642	0	2073								
				324		2126								
				0		941								
						0								

Table A.121: Routes of solution D – Part 1, 250_d2_tw3 ($Z2$ vs $Z5$).

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1959	451	1933	1471	322	430625	430014	2084	1508	430792	579	2134	1865	815	1688
430450	1116	2122	1781	430958	2165	430618	326	1888	1184	430804	1750	221	250	1001
856	974	1979	0	0	1890	1630	1987	968	0	1804	1173	315	2159	2031
1582	1873	645			2152	1601	1203	1448		0	1571	2107	1965	593
430793	0	2039			1777	1941	0	482			1166	1273	1917	1559
1707		1298			1938	0		1761			0	1900	0	1613
749		1102			0			0				0		0
1333		1948												
1450		1007												
0		0												

Table A.122: Routes of solution D – Part 2, 250_d2_tw3 ($Z2$ vs $Z5$).

V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
639	430761	486	2000	661	669	1729	306	1029	2003	1803	535	2048	1327	839
430389	430049	985	1162	300	1348	2008	1703	430155	453	430108	1678	1389	2136	838
1966	2091	1464	944	833	0	0	0	0	430141	1743	1808	430809	562	2145
1094	1928	0	1692	1717					1621	430749	1432	1618	491	0
1725	0		1879	1726					0	0	1482	0	0	
2081			1813	1877							1369			
236			765	553							0			
0			1602	0										
			0											

Table A.123: Routes of solution D – Part 3, 250_d2_tw3 ($Z2$ vs $Z5$).

V46	V47	V48	V49	V50	V51	V52	V53	V54	V55	V56	V57	V58	V59	V60
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1551	581	1721	1866	2104	1697	1925	2060	1677	1843	1384	1201	1714	1658	1526
192	430378	1588	1194	244	1219	1690	1524	312	1789	1240	1623	1028	1870	785
2097	0	1871	430806	1463	1961	0	1454	1897	1749	1937	0	1996	0	0
0		1872	0	430760	0		0	0	0	0		2149		
		2148		0								1235		
		0										2044		
												1875		
												0		

Table A.124: Routes of solution D – Part 4, 250_d2_tw3 ($Z2$ vs $Z5$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430017	579	250	1571	2145	815	1551	2100	2159	2044	2107	324	1777	2000	1987
838	1781	1294	1463	430618	1458	1877	1897	1601	1130	1890	430030	430049	430760	1049
1235	1203	1432	430465	2134	941	1725	1201	2084	0	326	0	0	430811	0
0	0	1197	0	1726	1162	0	1389	0		1576			765	
		430532		2008	1454		0			1344			593	
		1776		1464	1677					2073			1900	
		639		1813	1040					645			0	
		106		0	0					0				
		430749												
		0												

Table A.125: Routes of solution E – Part 1, 250_d2_tw3 ($Z4$ vs $Z3$).

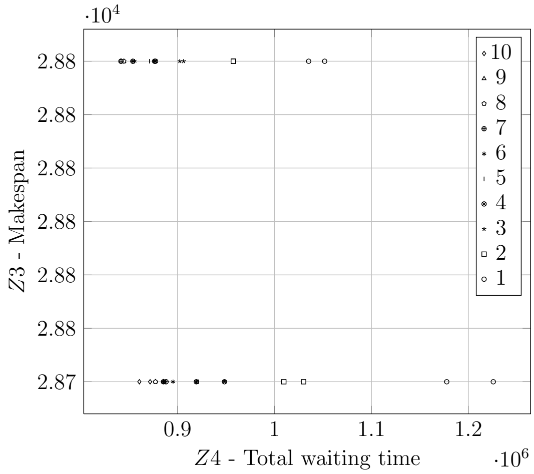

Figure A.177: Front progression of 250_d2_tw3 for $Z4$ vs $Z3$.

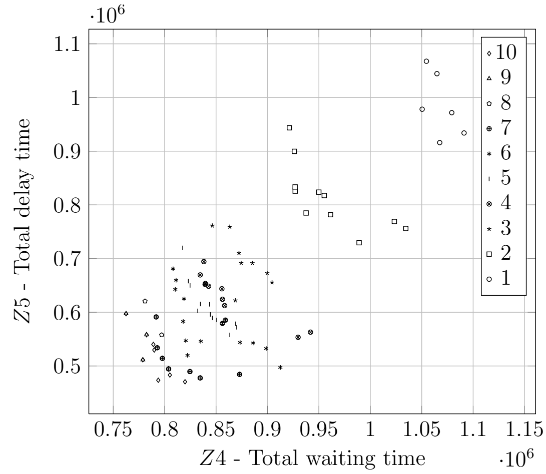

Figure A.178: Front progression of 250_d2_tw3 for $Z4$ vs $Z5$.

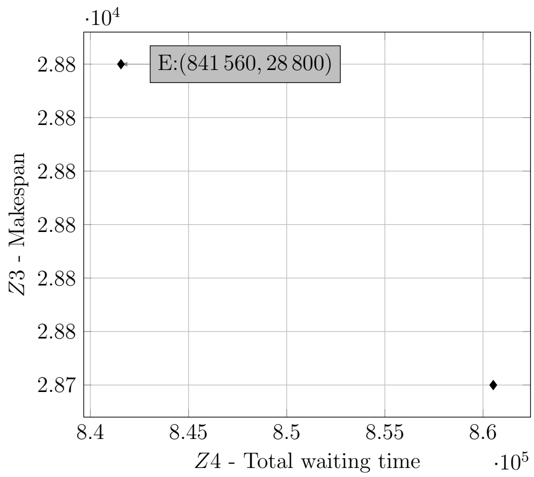

Figure A.179: Final approximation front of 250_d2_tw3 for $Z4$ vs $Z3$.

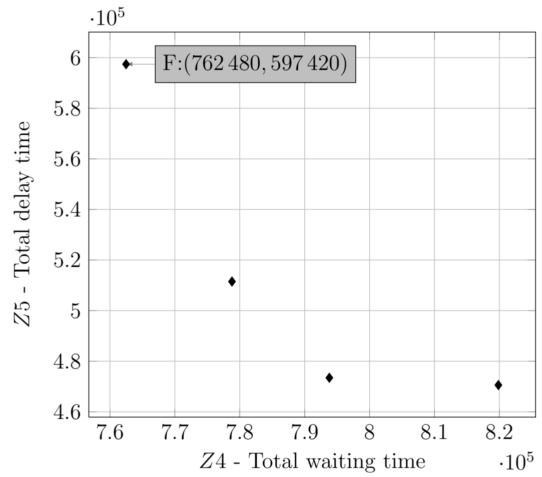

Figure A.180: Final approximation front of 250_d2_tw3 for $Z4$ vs $Z5$.

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1471	430148	1448	1996	1937	661	1526	1697	1621	1965	1873	1623	1707	1630	1327
1184	974	1678	430625	1583	1808	985	1686	1658	2165	924	1028	1584	1703	1679
0	2097	2060	1602	300	1789	430155	133	1240	483	430804	1729	669	0	1866
	0	2003	1495	1166	1369	1007	0	2048	430141	1875	322	1985		1865
		1610	1721	833	0	0		0	0	0	0	0		1524
		866	482	1582										430346
		1559	0	430809										968
		0		1688										0
				0										

Table A.126: Routes of solution E – Part 2, 250_d2_tw3 ($Z4$ vs $Z3$.)

V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1618	1717	312	221	1219	2126	1948	306	1933	1348	1867	2104	1938	430529	1635
749	430603	1173	1588	430659	1690	491	856	1671	948	430014	1094	430793	1528	451
486	430353	1692	430378	1941	0	1102	192	1298	1879	1769	785	1362	1888	1835
1872	1870	0	0	1333		0	236	430792	1619	1928	0	1757	944	1665
0	1116			0			0	2148	1979	430089		2091	1461	1029
	0							0	1843	1474		2031	1959	0
									0	0		0	1194	
													1804	
													0	

Table A.127: Routes of solution E – Part 3, 250_d2_tw3 ($Z4$ vs $Z3$.)

V46	V47	V48	V49	V50	V51	V52	V53	V54	V55	V56	V57	V58	V59	V60
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1743	430012	1803	430450	2138	2122	430108	839	1508	2149	453	535	315	907	1613
430278	1871	562	1750	2103	1140	2039	2136	1749	2152	1961	1917	2007	1509	1001
1482	0	2020	1966	1450	649	1384	1761	1925	1273	0	0	695	1714	430806
430389		0	553	430626	0	581	430958	0	0			0	430761	2121
0			430221	0		430471	0						1642	0
			0			2081							244	
						0							0	

Table A.128: Routes of solution E – Part 4, 250_d2_tw3 ($Z4$ vs $Z3$.)

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14
0	0	0	0	0	0	0	0	0	0	0	0	0	0
2000	1866	1873	1865	815	1804	1130	250	2107	1938	1613	1803	2149	839
1610	430017	430353	1508	1298	1813	1474	430806	315	430618	430155	1835	1571	669
1703	1870	2091	430809	1777	1029	2159	106	1879	1094	2103	430659	430465	1602
0	1184	1928	1235	1937	1140	1384	430532	866	1458	1362	948	1789	0
	0	0	0	0	1726	2152	645	430278	482	0	430141	0	
					0	974	924	491	1761		0		
						1618	2121	0	0				
						0	1389						
							0						

Table A.129: Routes of solution F – Part 1, 250_d2_tw3 ($Z4$ vs $Z5$.)

V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1769	1985	221	1776	579	1219	2084	1001	535	856	1601	1987	430761	1294	1630
785	1717	2031	2044	1697	661	430450	430049	453	1679	1941	486	430792	1482	1454
1900	2136	1576	244	322	1173	1948	2008	2126	1327	0	1677	1623	430221	2165
0	430346	1959	0	430378	941	1686	985	1690	1714		1450	0	0	1872
	1877	1559		0	2073	2097	2122	0	1642		0			0
	1464	1867			0	0	0		581					
	0	0							0					

Table A.130: Routes of solution F – Part 2, 250_d2_tw3 ($Z4$ vs $Z5$).

V30	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43
0	0	0	0	0	0	0	0	0	0	0	0	0	0
430014	1743	1583	430760	483	1965	1888	1635	1678	907	1979	306	944	430529
1495	1526	430389	430471	133	1890	1162	1348	553	2100	1528	1028	1582	430804
1463	649	1917	1588	430793	1194	2048	1925	1333	430625	1007	2039	1665	1875
1619	2020	0	1344	1961	1781	236	312	0	1166	1461	1707	1749	0
1432	0		0	0	0	2003	0	0	968	1996	324	639	
0						1551			1721	765	430030	0	
						1049			1725	1116	0		
						0			1369	0			
									0				

Table A.131: Routes of solution F – Part 3, 250_d2_tw3 ($Z4$ vs $Z5$.)

V30	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43
0	0	0	0	0	0	0	0	0	0	0	0	0	0
430014	1743	1583	430760	483	1965	1888	1635	1678	907	1979	306	944	430529
1495	1526	430389	430471	133	1890	1162	1348	553	2100	1528	1028	1582	430804
1463	649	1917	1588	430793	1194	2048	1925	1333	430625	1007	2039	1665	1875
1619	2020	0	1344	1961	1781	236	312	0	1166	1461	1707	1749	0
1432	0		0	0	0	2003	0	0	968	1996	324	639	
0						1551			1721	765	430030	0	
						1049			1725	1116	0		
						0			1369	0			
									0				

Table A.132: Routes of solution F – Part 4, 250_d2_tw3 ($Z4$ vs $Z5$.)

A.16 250_d2_tw4

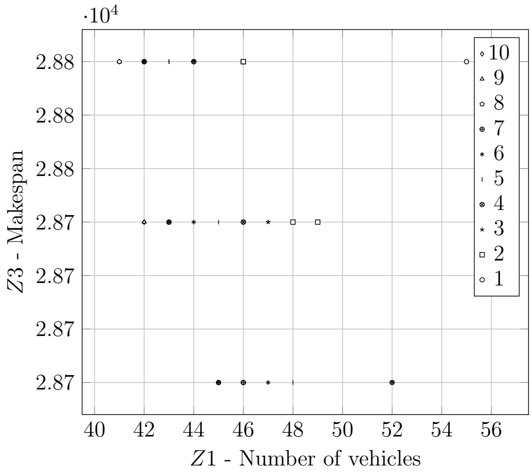

Figure A.181: Front progression of 250_d2_tw4 for $Z1$ vs $Z3$.

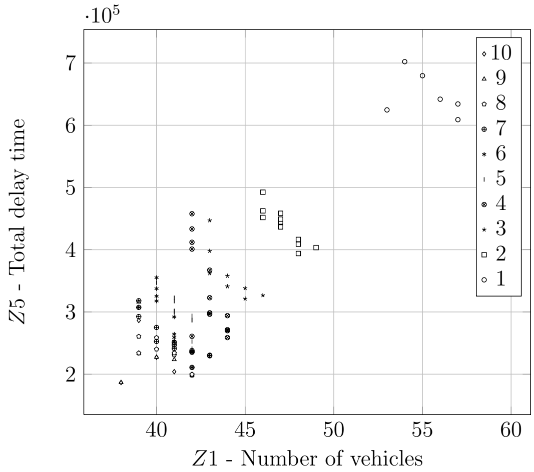

Figure A.182: Front progression of 250_d2_tw4 for $Z1$ vs $Z5$.

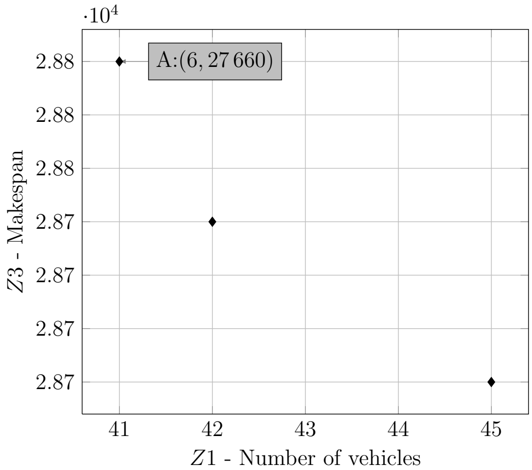

Figure A.183: Final approximation front of 250_d2_tw4 for $Z1$ vs $Z3$.

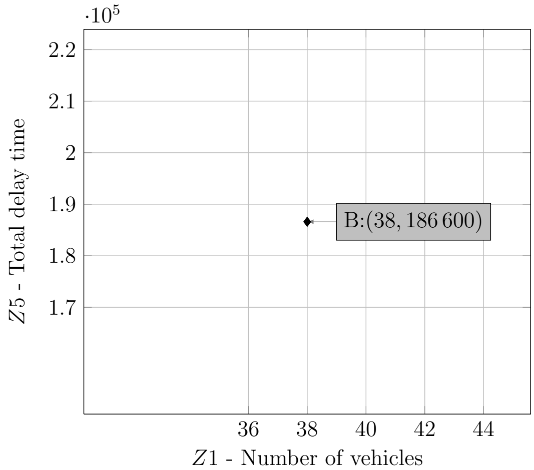

Figure A.184: Final approximation front of 250_d2_tw4 for $Z1$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1610	1623	430221	815	2149	1703	765	974	1865	324	1219	2107	785	430148
1613	430806	1130	2159	1551	1938	430108	315	322	1966	1875	221	2048	2039
482	1686	1584	2091	453	1235	1463	2031	1028	1890	2044	1621	1116	1448
839	430017	1294	838	0	2145	1344	1602	430625	1094	1509	192	907	300
1900	1749	649	0		1194	1877	1871	0	944	661	1761	1576	1601
0	0	1941			1348	1450	1872		1362	1384	2148	1678	2008
		430603			0	0	430389		1102	1458	0	1298	1162
		430278					2081		2007	562		1432	0
		1843					0		1726	430450		1808	
		1804							0	0		430760	
		1526										1642	
		244										2060	
		0										0	

Table A.133: Routes of solution A – Part 1, 250_d2_tw4 ($Z1$ vs $Z3$).

V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1559	430141	1690	1803	1813	553	695	1897	924	1917	1508	1671	1630	1743
1729	1571	1461	1203	1961	483	1750	430014	306	430089	2103	430618	1707	430378
2136	1166	2073	1965	326	1029	645	2152	430012	1928	312	1888	1769	985
430155	430793	1688	430792	593	1948	1464	1679	430761	0	1369	581	1583	1870
2134	430465	1866	669	0	1677	1197	1959	1389		2126	1721	0	579
133	2122	2100	0		0	856	0	1635		941	2000		1987
1327	1979	1528				430346		1184		430626	1776		0
1717	1454	1665				749		430811		0	430532		
2165	430471	486				1692		2104			0		
0	1879	833				2097		430958					
	106	0				236		0					
	639					0							
	0												

Table A.134: Routes of solution A – Part 2, 250_d2_tw4 ($Z1$ vs $Z3$).

V29	V30	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41
0	0	0	0	0	0	0	0	0	0	0	0	0
430030	1524	2084	1777	1173	1757	2121	535	1273	430804	1658	1240	968
1996	430809	1588	250	1781	1007	1873	430529	451	1001	1714	430049	1789
2003	1618	948	1201	1049	1835	430749	2020	0	1937	491	0	1985
1474	430659	430353	1471	0	1495	0	1725		0	866		1582
1933	0	1333	1697		1925		0			2138		1040
0		1619	1867		1140					0		0
		0	0		1482							
					0							

Table A.135: Routes of solution A – Part 3, 250_d2_tw4 ($Z1$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13
0	0	0	0	0	0	0	0	0	0	0	0	0
1897	1803	430141	1729	1835	2104	2121	1917	1941	1777	430378	1029	535
1690	430030	1001	430346	1094	645	2084	250	2073	221	1384	1658	1743
833	2159	1867	1583	2039	1007	1571	1865	1966	315	1749	312	483
968	2091	2149	1630	2100	430465	1996	1688	1240	1621	1201	482	430529
1166	326	2060	581	553	1040	1028	2145	1937	453	1761	1369	430618
430603	451	430811	2020	1588	1776	1804	2031	0	2003	192	1678	1273
974	0	430760	1619	430804	430532	0	1454		2148	593	0	2138
1194		0	0	430809	1130		1333		0	0		1474
1184				1618	1872		106					0
0				0	838		300					
					430659		0					
					0							

Table A.136: Routes of solution B – Part 1, 250_d2_tw4 ($Z1$ vs $Z5$).

V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26
0	0	0	0	0	0	0	0	0	0	0	0	0
1508	1173	1219	1813	1888	649	2107	1714	1870	1235	430749	430014	924
1610	765	430806	2000	1162	1203	1769	785	1938	749	0	1961	2136
1750	2044	1623	1362	1671	1965	1703	430089	2048	1576		1049	486
1933	324	1692	1432	1448	669	661	1928	1635	1509		0	1116
1526	2134	941	1197	430353	1781	1959	0	1987	866			430793
2103	1679	1450	1528	1879	1873	1757		839	0			2122
1877	2097	2007	944	1665	0	1707		0				1979
430958	1890	430471	1642	236		133						244
1602	0	1871	430450	0		1948						1344
1900		0	1808			1726						1458
0			0			0						1464
												0

Table A.137: Routes of solution B – Part 2, 250_d2_tw4 ($Z1$ vs $Z5$).

V27	V28	V29	V30	V31	V32	V33	V34	V35	V36	V37	V38
0	0	0	0	0	0	0	0	0	0	0	0
695	1298	985	1985	815	1843	1471	1482	579	856	306	1559
430626	1463	1461	1725	1495	430761	2126	430278	1789	430049	1686	1524
2152	1677	1584	430108	430792	639	0	491	0	1875	430017	430221
2008	430625	1613	430155	1551	1327		1925		430012	0	322
1582	1721	1866	1697	0	1717		2081		1389		0
0	430389	1294	1348		2165		1102		430148		
	0	1601	0		0		0		562		
		1140							907		
		948							0		
		0									

Table A.138: Routes of solution B – Part 3, 250_d2_tw4 ($Z1$ vs $Z5$).

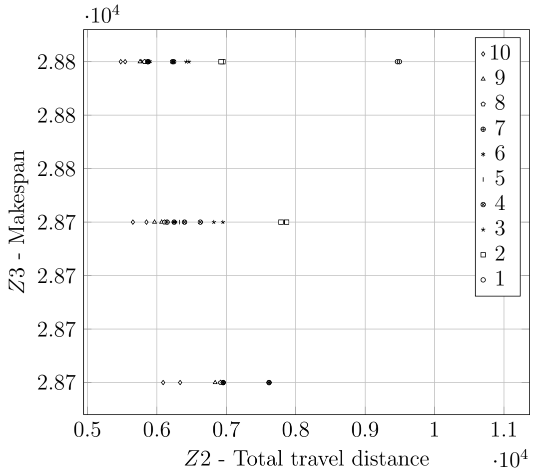

Figure A.185: Front progression of 250_d2_tw4 for $Z2$ vs $Z3$.

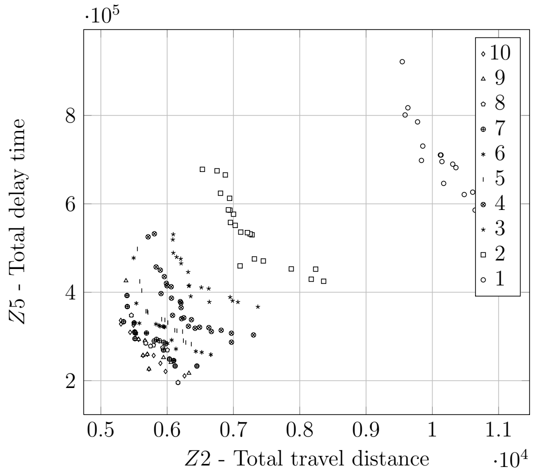

Figure A.186: Front progression of 250_d2_tw4 for $Z2$ vs $Z5$.

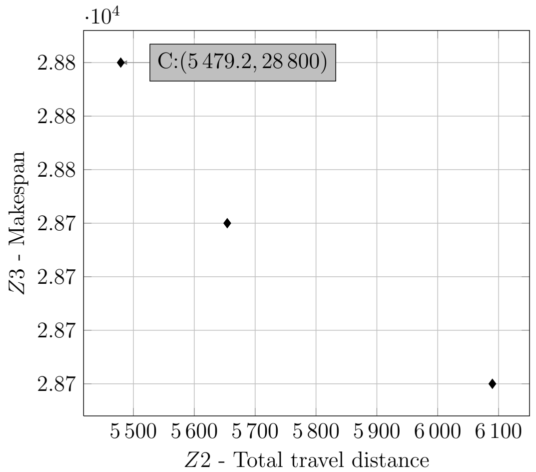

Figure A.187: Final approximation front of 250_d2_tw4 for $Z2$ vs $Z3$.

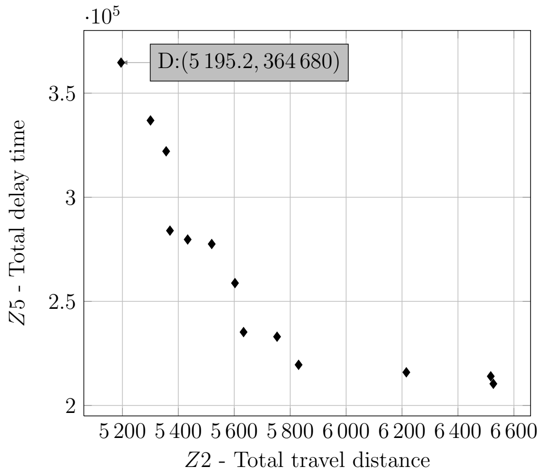

Figure A.188: Final approximation front of 250_d2_tw4 for $Z2$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1333	1482	1298	2145	649	306	1917	2044	430141	1559	1610	1729	695	430378	1777	2104
1726	1584	985	2048	1007	1873	483	2121	669	1471	944	1658	322	1721	326	948
236	1725	1707	1938	1461	430049	1703	1688	1551	430804	1162	1173	1714	1601	579	1678
1757	1571	1369	1866	1692	1240	430749	430030	430806	315	133	430811	1028	430017	0	2007
0	1448	1450	1865	645	1937	0	839	0	0	491	324	430221	2136		2103
	907	0	2149	1808	0		1867			1749	1813	0	0		0
	1463		1194	1582			430761			1040	244				
	486		1900	1528						300	0				
	856		0	1509						1102					
	430465			430353						866					
	106			0						1933					
	639									0					
	1776														
	2122														
	1464														
	0														

Table A.139: Routes of solution C – Part 1, 250_d2_tw4 ($Z2$ vs $Z3$).

V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30	V31
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
430148	1875	1671	312	1925	815	2008	1941	1789	2165	1879	553	1619	2091	974
430618	1613	1454	1677	1871	2159	2152	2039	1197	430389	1897	1495	430603	430659	192
1630	430792	562	1294	2148	1961	1362	1843	833	1583	1877	1679	2060	1623	1697
430529	838	2084	1094	1602	250	0	1690	749	1769	0	1959	0	0	593
535	1474	0	1130	430760	2138		968	2134	0		1717			0
1948	0		1872	0	0		1166	1508			1526			
2100			0				1835	1384			1804			
1979							1432	0			0			
0							1458							
							430625							
							941							
							0							

Table A.140: Routes of solution C – Part 2, 250_d2_tw4 ($Z2$ vs $Z3$).

V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45	V46
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1029	1524	1201	1588	1686	765	2073	430155	430014	924	1576	1870	430346	430012	1389
430450	1665	1987	1996	1888	1001	430471	1890	1635	1803	430793	1621	2020	451	430809
430626	430532	1928	2003	661	2107	1344	581	785	1219	430278	2031	1327	1985	0
2081	1140	1116	482	0	221	1750	1642	430089	1965	1203	1348	0	1761	
0	430958	0	0		1781	2000	2097	1184	1743	430108	1966		2126	
	0				1618	0	0	0	1273	1235	0		0	
					453				1049	0				
					0				0					

Table A.141: Routes of solution C – Part 3, 250_d2_tw4 ($Z2$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
306	1890	1925	2103	2107	430811	430603	1458	1201	1714	1173	1870	430221	815	1729
1917	1298	1872	1776	1985	553	1873	535	430809	1384	1897	765	1658	1961	1094
1703	1509	2148	106	221	430353	1697	430529	430792	968	2100	669	482	1273	1235
2091	1333	1900	1979	315	0	1184	1686	0	833	2084	430761	1551	0	1875
430659	1804	1526	1677	1867		2031	430017		1432	430793	1866	1116		2121
430806	1028	1871	2122	451		0	2000		907	1707	1865	0		974
0	244	0	2007	1761			944		430450	1454	1001			1688
	0		430532	1621			0		749	941	0			192
			1040	1294					1726	1464				430804
			0	1130					1166	0				2149
				0					0					0

Table A.142: Routes of solution D – Part 1, 250_d3_tw4 ($Z2$ vs $Z5$).

V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1559	430030	2008	1219	1813	695	430625	1843	2073	1482	326	1524	1749	1835	1508
430141	430089	1948	1743	2136	1588	1463	1162	1576	2145	579	312	985	2126	1671
1613	430049	1007	1203	1888	2097	639	1369	430278	839	0	1344	1879	1140	2152
1781	0	645	2159	1601	2165	0	1029	1966	838		1665	1789	1808	1362
1194		0	430749	1630	2081		1583	430346	1623		0	430389	0	1448
1348			0	1721	0		0	661	0			0		1678
0				0				1582						430465
								236						0
								430626						
								0						

Table A.143: Routes of solution D – Part 2, 250_d3_tw4 ($Z2$ vs $Z5$).

V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
866	1461	785	1928	1584	1102	856	2104	1803	649	430012	430378	1777	1690	1610
1495	2003	1965	1987	1769	300	2039	2060	483	430155	1471	1933	1938	430618	2134
430958	1996	2048	593	1941	1717	1750	324	250	562	1618	1877	1635	430148	430760
322	1474	1240	0	924	0	1757	948	430108	2020	453	1725	2044	1389	1450
491	0	1937		430014		1692	0	2138	0	0	1528	1049	133	1642
0		0		1959		1571		0			0	0	581	0
				1327		1619							0	
				1679		486								
				430471		1197								
				0		1602								
				0		0								

Table A.144: Routes of solution D – Part 3, 250_d3_tw4 ($Z2$ vs $Z5$).

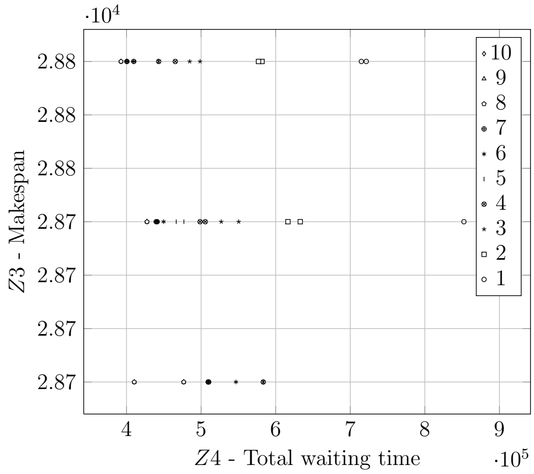

Figure A.189: Front progression of 250_d2_tw4 for $Z4$ vs $Z3$.

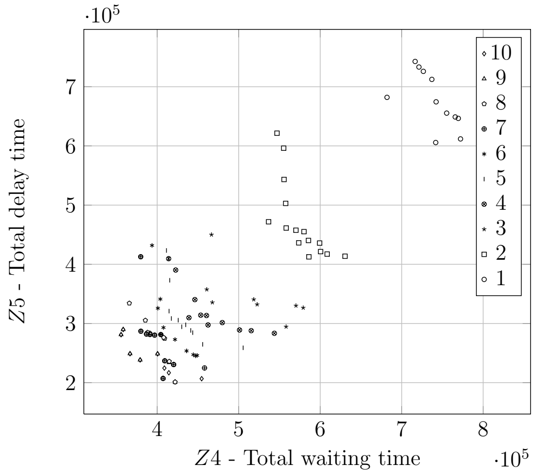

Figure A.190: Front progression of 250_d2_tw4 for $Z4$ vs $Z5$.

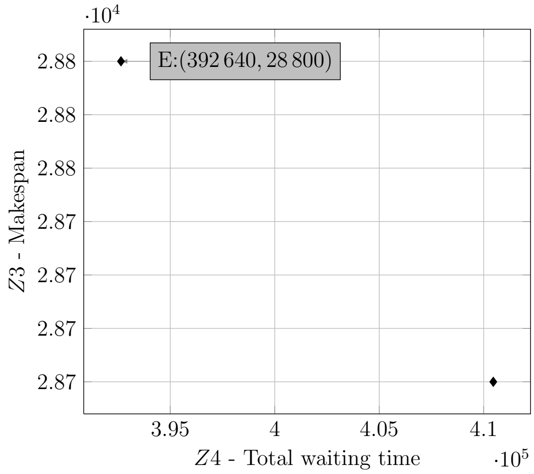

Figure A.191: Final approximation front of 250_d2_tw4 for $Z4$ vs $Z3$.

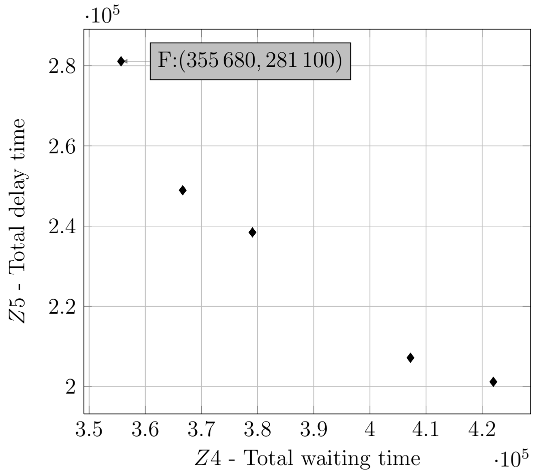

Figure A.192: Final approximation front of 250_d2_tw4 for $Z4$ vs $Z5$.

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1917	483	1866	649	1635	430221	430806	1725	2048	1559	1777	430346	985	1482
430378	1749	1865	221	2107	1959	1116	1621	562	968	1588	1495	1508	430532
1658	2091	1613	430014	315	2138	639	553	1007	1526	430792	1197	1890	2104
430603	1240	1966	2159	1743	1184	106	669	1933	1571	1474	2073	1679	451
2007	1937	430049	1697	486	0	1040	1348	1162	453	2165	1750	1877	765
1619	0	430749	430659	1789		1872	1618	0	430626	0	2000	2020	1721
1808		0	0	236		2148	0		1900		1464	749	2039
0				0		1623			0		1450	1166	430625
						322					1948	0	1327
						1028					1630		1726
						430389					2134		645
						1583					0		0
						0							

Table A.145: Routes of solution E – Part 1, 250_d2_tw4 ($Z4$ vs $Z3$).

V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28
0	0	0	0	0	0	0	0	0	0	0	0	0	0
430030	430148	1803	1965	306	1219	430529	1235	430012	1203	2044	1938	430141	1671
1688	815	581	1463	866	695	2084	1813	1769	1996	2145	1582	2136	1873
1601	312	833	1509	2152	1273	1690	1029	1471	838	1528	2060	1389	1875
430017	1781	430471	1454	1677	1049	1835	1665	1194	839	1717	324	1928	2149
1140	430804	1879	1298	1979	0	856	948	430761	0	661	430760	1729	2003
0	1761	1776	1576	0		430353	941	0		1384	2081	491	1871
	2031	0	1448			430811	1369			0	0	1925	0
	0		1432			1602	0					1678	
			0			1130						1692	
						0						430465	
												0	

Table A.146: Routes of solution E – Part 2, 250_d2_tw4 ($Z4$ vs $Z3$).

V29	V30	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41
0	0	0	0	0	0	0	0	0	0	0	0	0
2121	1201	1888	1173	785	1524	250	430618	1714	924	482	430278	1961
1686	1985	1870	326	192	974	1461	2103	1757	1094	430809	1584	1987
1843	430108	1941	579	1551	1703	1867	1707	430450	430089	0	1294	0
244	2122	1362	0	1804	1458	1001	1102	1642	0		133	
0	1344	2126		0	430958	593	300	0	0		1610	
	430155	1333			2097	2008	0				944	
	0	1897			2100	430793					535	
		0			0	907					0	
						0						

Table A.147: Routes of solution E – Part 3, 250_d2_tw4 ($Z4$ vs $Z3$).

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13
0	0	0	0	0	0	0	0	0	0	0	0	0
1750	535	1813	2048	1219	483	1917	553	1294	221	1938	1725	1524
856	1835	2039	1867	1584	430529	1870	1526	1941	1630	2107	2073	430014
1875	1966	974	1509	300	2103	1671	2084	1873	1686	2091	491	2159
1029	1635	1471	430809	1933	1757	866	1888	1890	1601	326	1495	1049
1781	1588	669	1618	1551	430465	1749	562	430389	430017	579	1959	0
1961	1344	1348	2149	453	1040	430353	2152	2081	0	0	312	
0	1692	0	0	1900	749	2060	1362	2165			1602	
	1102			0	581	1717	0	0			2148	
	0				1877	1464					1576	
											1432	
											0	

Table A.148: Routes of solution F – Part 1, 250_d2_tw4 ($Z4$ vs $Z5$).

V14	**V15**	**V16**	**V17**	**V18**	**V19**	**V20**	**V21**	**V22**	**V23**	**V24**	**V25**	**V26**
0	0	0	0	0	0	0	0	0	0	0	0	0
1688	1482	430141	430148	1777	1001	1865	1703	924	1389	2136	1769	815
430761	1173	985	645	944	1559	1866	430089	430346	1235	430012	1201	1610
315	1116	695	430532	2000	250	1184	1937	1571	430793	1665	661	430221
1508	2031	1162	486	2008	324	430749	0	430278	133	430471	2145	1979
2134	1474	1729	1658	1007	430760	0		1140	1583	1678	430155	1707
1327	1872	1690	1623	1450	1642			907	1761	833	1948	430450
1028	1130	2100	451	941	1333			1166	1621	968	2122	948
236	1871	430811	1789	0	430958			1726	0	1808	0	0
0	1804	1582	322		0			0		0		
	2003	1528	639									
	244	0	0									
	1776											
	0											

Table A.149: Routes of solution F – Part 2, 250_d2_tw4 ($Z4$ vs $Z5$).

V27	**V28**	**V29**	**V30**	**V31**	**V32**	**V33**	**V34**	**V35**	**V36**	**V37**	**V38**	**V39**
0	0	0	0	0	0	0	0	0	0	0	0	0
1714	1743	430049	306	1094	1897	1613	1965	2104	1803	1203	785	765
1843	1384	1240	430108	1721	430603	1461	430625	430378	1985	430030	1619	1454
430804	430618	1928	2020	1677	1273	1448	1463	649	1298	482	2097	0
1987	1996	0	1879	2007	2138	1197	430626	1458	2121	1194	1925	
430806	2044		106	2126	593	1679	0	839	430792	0	0	
0	1697		0	0	0	838		192	430659			
	0					0		1369	0			
								0				

Table A.150: Routes of solution F – Part 3, 250_d2_tw4 ($Z4$ vs $Z5$).

APPENDIX B

SELECTION OF RESULTS OF THE WPBTS BLOOD INVENTORY MANAGEMENT PROBLEM

This appendix documents a selection of results for the WPBTS blood inventory management problem. Table B.1 documents the limits used in the CEM algorithm for the variables (combinations of blood products, blood groups and blood banks). Values are given for the minimum and maximum values for both the ideal inventory and reorder points. The current ideal inventory levels are also given. The rest of the appendix documents some results from the case study. Final approximation fronts as obtained by the cross-entropy method for multi-objective optimisation are shown in **Chapter 5** with one point per front selected. This point in the objective function space is represented here with a table documenting the decision variables at that particular point. The decision variables represent the suggested inventory policy at the various blood banks.

Product details			**Inventory**			**Reorder point**	
			Max	**Min**	**Current**	**Max**	**Min**
GEO	**WB**	**O^-**	19	0	4	16	0
	RCC	**O^+**	121	41	71	111	31
		O^-	35	5	15	28	2
		A^+	105	25	55	95	18
		A^-	20	0	10	18	0
		B^+	75	15	35	65	10

Continued on next page

Product details			Inventory			Reorder point	
			Max	Min	Current	Max	Min
		B$^-$	16	0	5	14	0
		AB$^+$	24	4	9	19	1
	LDRC	**O$^+$**	40	0	10	33	0
		O$^-$	12	0	0	10	0
		A$^+$	40	0	10	33	0
		A$^-$	8	0	4	6	0
		B$^+$	14	0	4	12	0
GSH	**WB**	**O$^+$**	33	3	8	28	0
		O$^-$	17	0	2	14	0
	RCC	**O$^+$**	130	50	80	120	40
		O$^-$	27	0	10	23	0
		A$^+$	115	35	65	105	25
		A$^-$	8	0	7	5	0
		B$^+$	78	18	38	68	13
		B$^-$	6	0	4	5	0
		AB$^+$	42	2	12	35	0
		AB$^-$	4	0	2	3	0
	LDRC	**O$^+$**	100	20	50	90	13
		O$^-$	21	1	6	16	0
		A$^+$	70	10	30	60	5
		A$^-$	16	0	4	14	0
		B$^+$	31	1	6	26	0
		B$^-$	11	0	2	9	0
MCV	**RCC**	**O$^+$**	85	15	40	75	10
		O$^-$	20	0	5	15	0
		A$^+$	75	15	35	65	10
		A$^-$	22	2	7	17	0
		B$^+$	32	2	12	25	0
		B$^-$	9	0	3	7	0
		AB$^+$	16	0	4	14	0
	LDRC	**O$^+$**	35	5	15	28	2
		O$^-$	18	0	5	15	0
		A$^+$	45	5	15	38	2
		A$^-$	10	0	4	8	0
		B$^+$	16	0	4	14	0
		B$^-$	7	0	2	5	0
PAARL	**RCC**	**O$^+$**	76	16	36	66	11

Continued on next page

Product details			Inventory			Reorder point	
			Max	Min	Current	Max	Min
		O^-	19	1	6	16	0
		A^+	61	11	26	51	6
		A^-	8	0	4	6	0
		B^+	43	3	13	36	0
		B^-	7	0	2	5	0
		AB^+	23	0	4	18	0
	LDRC	O^+	9	0	4	7	0
		A^+	8	0	4	7	0
		B^+	6	0	2	5	0
RCX	WB	O^+	4	0	4	3	0
		O^-	5	0	2	4	0
		A^+	2	0	2	1	0
		A^-	17	0	2	14	0
		B^+	3	0	1	2	0
		AB^+	16	0	1	13	0
	RCC	O^+	90	20	45	80	13
		O^-	21	1	6	16	0
		A^+	70	10	30	60	5
		A^-	14	0	5	12	0
		B^+	59	9	24	49	6
		B^-	3	0	2	2	0
		AB^+	24	0	4	21	0
		AB^-	7	0	2	5	0
	LDRC	O^+	59	9	24	49	6
		O^-	40	0	10	33	0
		A^+	57	7	22	47	4
		A^-	16	3	8	14	0
		B^+	42	2	12	35	0
		B^-	8	0	3	6	0
TBH	WB	O^+	17	0	2	14	0
		O^-	19	0	4	16	0
	RCC	O^+	160	80	110	150	70
		O^-	41	5	15	38	2
		A^+	145	65	95	135	55
		A^-	20	2	12	18	0
		B^+	90	20	45	80	13
		B^-	8	0	5	6	0

Continued on next page

Product details			Inventory			Reorder point	
			Max	Min	Current	Max	Min
		AB$^+$	45	5	15	38	2
		AB$^-$	4	0	2	2	0
	LDRC	**O$^+$**	70	10	30	60	5
		O$^-$	20	0	5	15	0
		A$^+$	45	5	15	38	2
		A$^-$	20	0	5	15	0
		B$^+$	40	0	10	33	0
		B$^-$	10	0	2	8	0
WTR	**RCC**	**O$^+$**	101	21	51	91	14
		O$^-$	10	2	7	8	0
		A$^+$	71	11	31	61	6
		A$^-$	10	0	10	8	0
		B$^+$	60	10	25	50	5
		B$^-$	3	0	2	2	0
		AB$^+$	18	2	7	16	0
	LDRC	**O$^+$**	14	0	4	11	0
		A$^+$	12	0	4	10	0

Table B.1: Summary of limits used in the MOO CEM for the WPBTS case study.

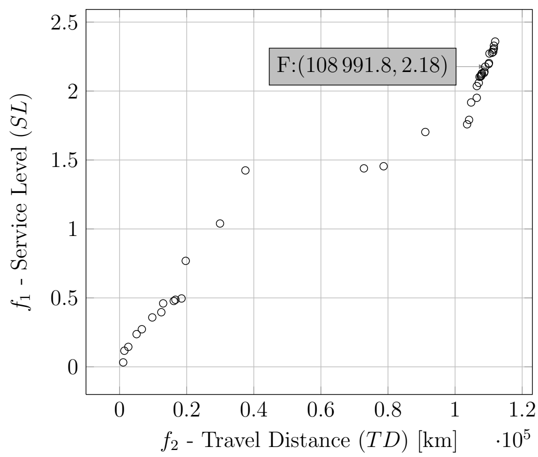

Figure B.1: Final approximation front for system with a modelled constrained supply source at 75 %.

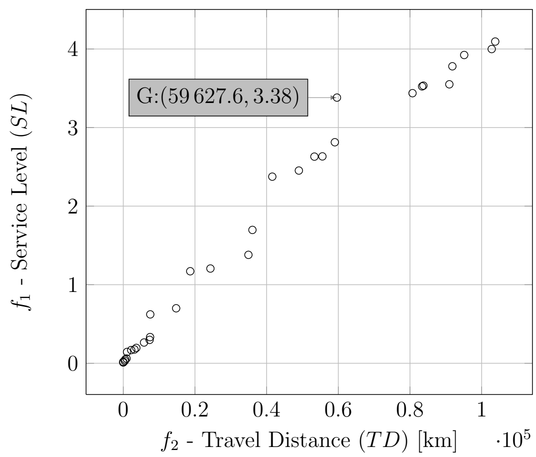

Figure B.2: Final approximation front for system with a modelled constrained supply source at 125%.

		O⁺		**O⁻**		**A⁺**		**A⁻**		**B⁺**		**B⁻**		**AB⁺**		**AB⁻**	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			10	10												
	GSH	25	10	9	4												
	MCV																
	PAARL																
	RCX	1	1	2	1	2	0	9	1	1	1			3	1		
	TBH	11	1	4	0												
	WTR																
RCC	**GEO**	63	62	25	3	40	40	10	10	42	29	11	7	18	4		
	GSH	104	54	4	0	44	41	3	2	50	23	5	3	41	20	0	0
	MCV	66	22	13	5	42	35	20	15	25	0	3	3	9	0		
	PAARL	75	27	19	13	30	26	1	0	24	15	3	1	12	10		
	RCX	62	61	2	2	16	14	1	0	50	49	3	0	10	9	4	1
	TBH	96	95	22	2	127	123	8	4	33	32	3	0	19	10	0	0
	WTR	69	62	7	7	49	21	7	5	47	18	0	0	11	0		
LDRC	**GEO**	13	10	8	8	31	0	6	6	8	7						
	GSH	56	54	13	12	41	31	9	0	6	4	8	3				
	MCV	27	13	16	6	27	3	5	4	0	0	4	1				
	PAARL	0	0			3	1			3	1						
	RCX	22	12	7	3	41	28	13	1	35	22	3	2				
	TBH	10	9	17	16	27	2	6	1	37	0	0	0				
	WTR	14	7			10	3										

Table B.2: Values of variables for solution F (system with modelled constrained supply source at 75%, SL vs TD).

		O$^+$ Inv	O$^+$ ROP	O$^-$ Inv	O$^-$ ROP	A$^+$ Inv	A$^+$ ROP	A$^-$ Inv	A$^-$ ROP	B$^+$ Inv	B$^+$ ROP	B$^-$ Inv	B$^-$ ROP	AB$^+$ Inv	AB$^+$ ROP	AB$^-$ Inv	AB$^-$ ROP
WB	**GEO**			10	0												
	GSH	10	0	4	1												
	MCV																
	PAARL																
	RCX	3	2	1	0	8	0	5	0	3	3			6	3		
	TBH	9	5	12	0												
	WTR																
RCC	**GEO**	5	0	10	2	21	2	9	2	15	0	10	5	0	0		
	GSH	15	0	5	1	3	0	3	0	17	11	7	3	4	2	9	3
	MCV	10	0	7	4	14	11	9	3	5	1	15	2	11	2		
	PAARL	10	6	17	5	7	0	8	3	11	4	3	2	4	1		
	RCX	6	4	3	3	10	5	7	1	10	6	0	0	3	2	7	1
	TBH	44	25	6	6	26	6	10	2	7	7	11	5	25	4	3	0
	WTR	11	7	9	3	20	6	5	0	13	3	5	0	3	1		
LDRC	**GEO**	1	0	3	0	26	5	17	2	9	0						
	GSH	27	17	14	1	2	0	4	1	4	3	6	2				
	MCV	14	1	11	0	10	6	10	0	8	4	1	1				
	PAARL	7	2			5	1			4	0						
	RCX	17	1	19	1	13	7	4	2	5	3	3	2				
	TBH	8	6	1	0	1	0	3	3	6	0	9	3				
	WTR	12	0			2	0										

Table B.3: Values of variables for solution G (system with modelled constrained supply source at 125%, SL vs TD).

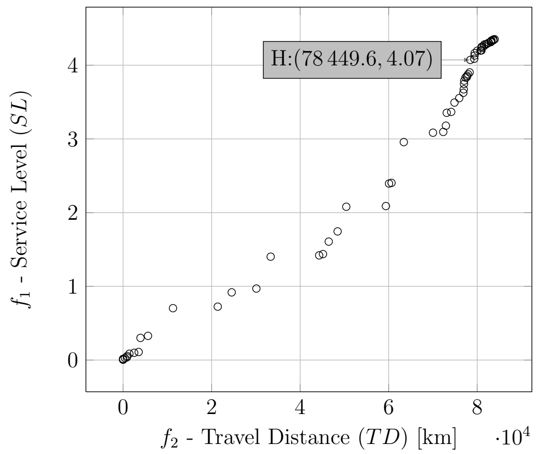

Figure B.3: Final approximation front for system with a modelled perfect supply source ($c_1 = 0.01, c_2 = 0.5$).

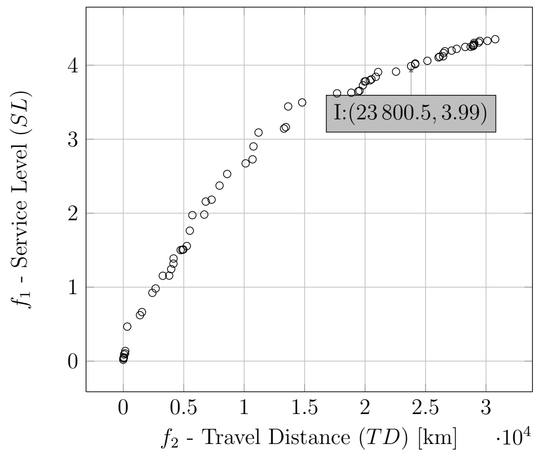

Figure B.4: Final approximation front for system with a modelled perfect supply source ($c_1 = 0.01, c_2 = 0.01$).

		O⁺		O⁻		A⁺		A⁻		B⁺		B⁻		AB⁺		AB⁻	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			10	3												
	GSH	14	13	5	5												
	MCV																
	PAARL																
	RCX	2	0	4	2	0	0	6	3	0	0			7	3		
	TBH	12	11	15	4												
	WTR																
RCC	**GEO**	43	43	16	12	46	26	14	13	40	20	15	4	13	13		
	GSH	87	77	19	18	56	56	6	1	38	38	3	3	23	21	0	0
	MCV	61	41	8	6	56	32	14	4	26	14	5	0	10	8		
	PAARL	50	48	9	9	43	40	7	2	23	22	4	4	11	10		
	RCX	62	49	11	5	67	52	4	4	37	27	0	0	15	10	7	3
	TBH	115	101	15	12	98	89	7	3	63	44	6	2	14	13	2	0
	WTR	50	50	7	7	45	31	7	7	38	28	0	0	11	8		
LDRC	**GEO**	39	10	8	8	27	27	1	0	9	8						
	GSH	73	53	16	14	40	29	4	3	20	10	4	4				
	MCV	29	11	13	4	25	22	4	4	9	4	5	3				
	PAARL	6	3			6	0			4	3						
	RCX	39	27	23	15	51	32	14	14	39	17	4	3				
	TBH	32	16	8	6	37	12	16	9	23	19	10	2				
	WTR	4	1			6	1										

Table B.4: Values of variables for solution H (system with modelled perfect supply source, SL vs TC $(c_1 = 0.01, c_2 = 0.5)$).

		O⁺		O⁻		A⁺		A⁻		B⁺		B⁻		AB⁺		AB⁻	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			18	8												
	GSH	23	10	12	7												
	MCV																
	PAARL																
	RCX	2	0	2	2	0	0	8	7	0	0			12	9		
	TBH	9	6	18	13												
	WTR																
RCC	**GEO**	67	49	30	28	67	0	10	9	43	37	8	8	11	0		
	GSH	95	58	24	21	75	52	4	4	57	30	4	2	7	7	0	0
	MCV	41	40	8	8	56	50	17	9	25	13	6	1	14	13		
	PAARL	40	40	7	7	40	28	4	1	31	21	6	4	9	9		
	RCX	46	19	13	6	63	17	6	4	16	14	0	0	15	7	5	4
	TBH	80	78	33	8	105	65	14	11	43	36	0	0	40	8	3	0
	WTR	42	39	2	1	52	46	2	2	34	32	0	0	11	10		
LDRC	**GEO**	21	21	5	1	17	17	5	2	12	6						
	GSH	35	34	16	9	52	31	7	3	12	2	11	2				
	MCV	17	12	16	4	14	12	6	6	6	5	4	4				
	PAARL	8	5			2	2			3	3						
	RCX	32	18	31	15	53	32	5	5	16	16	8	4				
	TBH	44	31	20	12	18	6	15	10	28	21	10	8				
	WTR	8	5			11	7										

Table B.5: Values of variables for solution I (system with modelled perfect supply source, SL vs TC $(c_1 = 0.01, c_2 = 0.01)$).

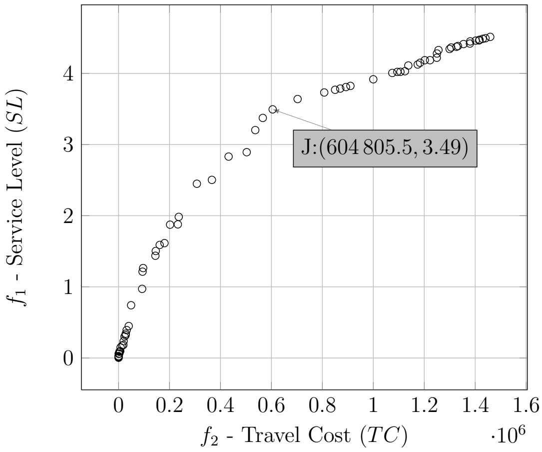

Figure B.5: Final approximation front for system with a modelled perfect supply source ($c_1 = 0.5, c_2 = 0.01$).

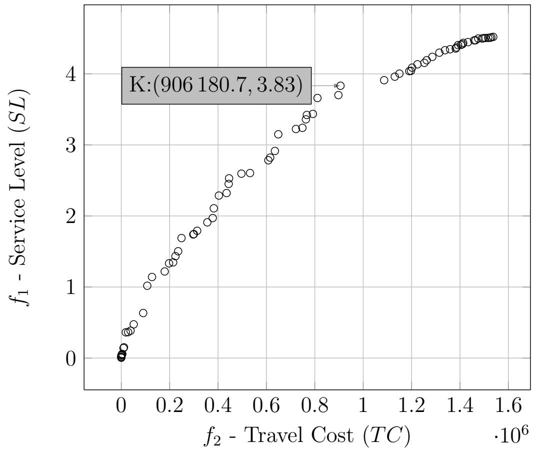

Figure B.6: Final approximation front for system with a modelled perfect supply source ($c_1 = 0.5, c_2 = 0.5$).

		O^+		O^-		A^+		A^-		B^+		B^-		AB^+		AB^-	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			8	8												
	GSH	12	11	7	2												
	MCV																
	PAARL																
	RCX	17	0	8	4	6	5	11	2	4	3			7	6		
	TBH	8	1	7	6												
	WTR																
RCC	**GEO**	33	0	16	8	2	2	8	7	19	19	6	3	1	0		
	GSH	48	45	18	11	33	27	9	7	36	33	13	2	26	7	1	0
	MCV	21	6	4	4	27	27	14	6	9	8	9	6	13	1		
	PAARL	24	14	9	2	41	16	3	3	9	2	2	0	12	1		
	RCX	46	22	9	5	44	31	7	6	31	17	13	1	5	4	14	14
	TBH	36	28	26	7	46	24	22	5	35	22	1	1	16	12	7	1
	WTR	20	0	7	7	34	21	21	9	26	13	8	1	10	9		
LDRC	**GEO**	13	13	7	4	26	13	10	2	11	3						
	GSH	43	34	3	3	33	8	7	2	8	2	12	0				
	MCV	10	1	14	12	20	9	9	9	6	3	10	0				
	PAARL	5	3			2	1			3	1						
	RCX	20	7	21	5	26	18	20	3	16	14	16	0				
	TBH	41	14	9	3	9	9	10	0	17	11	6	6				
	WTR	7	5			2	2										

Table B.6: Values of variables for solution J (system with modelled perfect supply source, SL vs TC ($c_1 = 0.5, c_2 = 0.01$)).

		O^+		O^-		A^+		A^-		B^+		B^-		AB^+		AB^-	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			9	4												
	GSH	4	4	14	10												
	MCV																
	PAARL																
	RCX	13	8	7	6	16	9	4	3	3	3			8	6		
	TBH	6	6	9	5												
	WTR																
RCC	**GEO**	15	15	14	4	51	0	15	2	30	8	9	7	6	6		
	GSH	70	67	21	6	37	22	10	2	46	19	5	5	19	14	11	8
	MCV	50	23	6	4	34	24	13	6	11	11	9	6	5	3		
	PAARL	45	20	18	5	29	15	5	2	26	21	6	4	12	1		
	RCX	44	22	12	9	34	26	6	6	36	34	8	8	4	2	8	4
	TBH	92	80	17	12	55	55	22	7	32	18	7	7	14	14	5	5
	WTR	9	9	12	3	46	21	18	6	19	14	11	9	16	6		
LDRC	**GEO**	9	9	2	2	15	4	7	6	1	0						
	GSH	66	50	8	3	33	30	9	9	12	11	8	0				
	MCV	14	14	5	0	31	25	5	5	5	2	13	6				
	PAARL	12	9			18	9			3	2						
	RCX	15	15	29	2	25	14	10	1	10	10	16	3				
	TBH	43	18	19	5	35	27	15	3	20	20	8	2				
	WTR	9	8			6	2										

Table B.7: Values of variables for solution K (system with modelled perfect supply source, SL vs TC $(c_1 = 0.5, c_2 = 0.5)$).

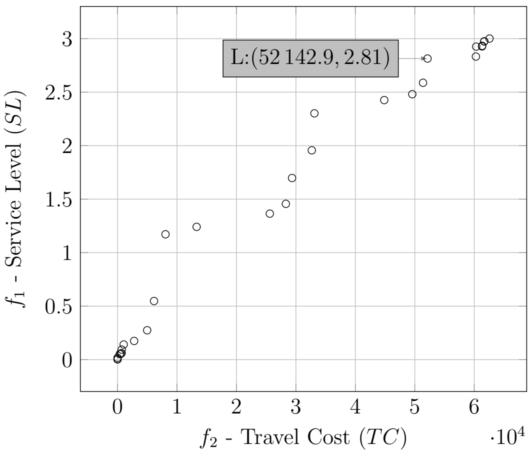

Figure B.7: Final approximation front for system with a modelled constrained supply source $(c_1 = 0.01, c_2 = 0.5)$.

		O^+		O^-		A^+		A^-		B^+		B^-		AB^+		AB^-	
		Inv	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**	**Inv**	**ROP**
WB	**GEO**			0	0												
	GSH	0	0	4	3												
	MCV																
	PAARL																
	RCX	2	0	0	0	0	0	1	0	0	0			5	4		
	TBH	4	4	5	1												
	WTR																
RCC	**GEO**	42	0	4	3	7	0	6	0	13	2	3	0	5	3		
	GSH	5	4	1	1	27	6	3	1	4	0	0	0	10	1	0	0
	MCV	19	7	2	2	19	6	4	1	7	2	2	1	2	1		
	PAARL	31	9	7	6	14	3	0	0	6	5	0	0	1	1		
	RCX	15	15	8	5	4	4	2	2	10	1	0	0	1	0	1	1
	TBH	58	8	16	2	43	26	3	2	7	7	1	0	20	2	4	0
	WTR	49	4	2	2	35	1	0	0	10	1	0	0	1	0		
LDRC	**GEO**	1	1	1	1	0	0	2	0	2	0						
	GSH	26	14	7	2	11	2	5	5	6	3	0	0				
	MCV	12	1	8	3	21	2	1	0	5	1	1	0				
	PAARL	3	3			2	1			2	2						
	RCX	5	5	5	0	29	0	7	1	12	4	0	0				
	TBH	4	2	0	0	1	0	2	2	0	0	4	0				
	WTR	2	0			5	4										

Table B.8: Values of variables for solution L (system with modelled constrained supply source, SL vs TC ($c_1 = 0.01, c_2 = 0.5$)).

APPENDIX C

WPBTS CONFIDENTIALITY DOCUMENT

WP Blood Transfusion Service
Incorporated Association Not For Gain | Ingelyfde Vereniging Sonder Winsoogmerk
WP Bloedoortappingsdiens

Head Office:
Old Mill Road, Pinelands, 7405
PO Box 79, Howard Place 7450
info@wpbts.org.za
T: 021 507 6300
F: 021 531 0322

Directorate:
F: 021 531 4884
Fractionation:
101 Connaught Rd
Beaconvale, Parow 7500
T: 021 933 9400
F: 021 931 5551

George:
Medical Centre, Courtenay Street
PO Box 65, George 6530
T: 044 874 2074
F: 044 874 6097

Paarl:
263 Main Road, Paarl 7646
PO Box 422, Paarl 7620
T: 021 871 1030
F: 021 872 5945

Worcester:
26 Napier Street, Worcester 6850
PO Box [illegible], Worcester 6849
T: 023 342 2450
F: 023 342 7556

www.wpblood.org.za

17 May 2012

To Whom It May Concern:

I, Charlotte Hauman, a student of Industrial Engineering, University of Stellenbosch, understand that confidentiality and high ethical standards are important prerequisites to a successful research case study, a study that involves access to confidential information about the company.

I undertake to abide by the following:

1. To keep confidential, information gained about individual blood donors.
2. To keep confidential, personal information gained about WPBTS staff, particularly information relating to individual salaries.
3. To respect any requests by WPBTS to protect their relationship with third party vendors by not revealing specific amounts paid to those vendors for certain products or services. Where necessary and appropriate, figures will be modified so as to not reveal actual amounts.
4. To discuss interim findings with WPBTS and provide WPBTS with a copy of the research report for their approval regarding sensitive or confidential information, prior to final submission and publication.
5. Not to use information learned about WPBTS for personal gain, directly or indirectly.

I acknowledge that these obligations continue after completion of the ~~MBA~~ course work.

NAME: CHARLOTTE HAUMAN

SIGNATURE:

DATE: 22/05/2012

Pr. No: 7800045
Pr. No: 052 000 0191183
Reg. No: 1943/016692/08
Directors: GRM Bellairs; AR Bird; GR Bosman; F Essop; BdL Figaji; I Kaprey; PJ Morris; N Parker (Chairman); CRB Prior; R Ramsbottom; PK Slack; E Steyn PBR07 (27 Nov10)